

NORTHWESTERN UNIVERSITY

iShed:
Interactive Software for the
Training of Novice Jazz Improvisers

A DISSERTATION

SUBMITTED TO THE GRADUATE SCHOOL
IN PARTIAL FULFILLMENT OF THE
REQUIREMENTS

for the degree

DOCTOR OF PHILOSOPHY

Field of Music Studies

By

Marc Jacoby

EVANSTON, ILLINOIS

December 2006

ABSTRACT

iShed:
Interactive Software for the
Training of Novice Jazz Improvisers

It's quite common for the novice musician to forget their place in the song, miss the change to a new section, or be unable to play in the key or at the tempo chosen by the leader. Every type of improvisational playing requires the player to follow a set of rules, to understand the particular style for its "codes of behavior" so they may develop an appropriate strategy for successful navigation.

Where can young or novice jazz musicians get the training needed to become more effective improvisers? Research suggests that quite often in music education, even when a jazz band program exists, training in improvisation is not a high priority because of the time and dedication the student needs to commit in order to master the skills required, or because of the music teacher's unfamiliarity with the subject.

The purpose of this dissertation was to design software is to provide a highly interactive practice environment for skill acquisition in the domain of jazz improvisation. The activities available via the software interface reflect a synthesis of methodologies articulated in jazz improvisation textbooks and personal communications with jazz masters. *iShed* attempts to put into practice a belief that music education, specifically jazz improvisation, can be made more dynamic and engaging to the novice player through the use of computer technology.

ACKNOWLEDGEMENTS

I start this process of acknowledging all those who have been instrumental to completing this dissertation with my parents Ben and Martha, who had to deal with the possibility that I would not graduate from high school because I couldn't keep a clean notebook in health class. Thanks for your endless support. I love you both;

Scott Lipscomb, my advisor, for his guidance, investment in time, friendship, and never-ending encouragement that I could really complete such a project. Only you had the expertise in research, multimedia, programming, and jazz that could help me see this through. You helped to keep me on track, making sure I didn't succumb to "feature creep;"

The faculty in the Music Education department at Northwestern University – Peter Webster and Maud Hickey (dissertation committee members), Carlos Abril, Janet Barrett, Bennett Reimer, and Bernard Dobroski- as well as Romona Wis, Stephen Hill, and Daniel Edelson. You've been fine teachers and role models of what the academy can be;

The other doctoral students I've had the good fortune getting to know. I value your friendship;

My "brother" Richard Pillot. Only you can yell at me on the bandstand;

My uncle Harry Jacoby, who taught me my first swing ride pattern on the drumset;

My new colleagues at West Chester University. Thanks for the words of encouragement;

Finally, I could not have finished this dissertation without the support of my wife Roxanne, and children Rachel and Ellis. If love is never having to say you're sorry, then I can only thank you for putting up with me over the past year. I love you all.

TABLE OF CONTENTS

CHAPTER 1 – INTRODUCTION AND THEORETICAL FOUNDATIONS	8
Why Create <i>iShed</i>	11
Connection to CSEME	13
Summary	14
Chapter Organization	15
 CHAPTER 2 – REVIEW OF THE LITERATURE	 17
Jazz Improvisation.....	17
Jazz Improvisation and Education	22
Computers in Education	25
Computer Aided Instruction (CAI)	27
CAI Efficacy	28
User Interface.....	31
Review of Current Applications	33
SmartMusic	34
Sheddin’ the Basics – Jazz Piano	35
Sheddin’ the Basics – Latin Jazz.....	35
Jazz Piano/Guitar Masterclass.....	36
Century of Jazz Piano	36
Band in a Box / MIBAC Jazz.....	37
Summary.....	38
 CHAPTER 3 –SOFTWARE DEVELOPMENT	 40
Description of the Software	40
Software Initialization.....	41
Jazz Club	41
The Woodshed.....	44
PatternLoader.....	44
RiffBuilder.....	47
Groove’ranger.....	49
Ear Training	52
Transcriber	52
RhythmBuilder.....	54
The Record Store.....	55
Design Process	56
Repertoire Development	56
Pattern/Motivic Skill Acquisition.....	56
Aural Skills.....	58
Chord Types and Harmonic Progression	58

Development of <i>iShed</i>	60
Development Environment	60
Flash to Director Communication	61
Special Classes (Xtras)	62
Sequence Xtra	63
AmiXtra	63
BuddyAPI	63
Core Engine Development	63
Notation to MIDI	64
The 5ths Wheel	65
Chunk Arranger	66
Parse MIDI Data	67
MIDI and Transpose MIDI Tracks	68
Transport Controls	69
Mixing Controls	70
Audio Recording with MIDI	70
Summary	71
CHAPTER 4 – OPPORTUNITIES, DIFFICULTIES, AND CONCLUSIONS	72
Innovation and Enhanced Capabilities of Interactive Multimedia	72
Impact on Performance Training	72
Strengths of the Software	73
Assessment	75
Outstanding Issues	76
Recommendations for Research	78
Conclusion	79
REFERENCES	82

CHAPTER 1 – INTRODUCTION AND THEORETICAL FOUNDATIONS

iShed Software

iShed is a software program for the Windows or Macintosh operating systems that provides a highly interactive practice environment for skill acquisition in the domain of jazz improvisation. This paper accompanies the submission of the software providing a rationale for and explanation of its development. Before reading this document, it is recommended that you download and explore the program. A freeware version can be found on the World Wide Web at <http://www.roxmedia.com/jacoby/ished>.

Introduction

Some time ago, a young jazz improviser, barely 15 years old, experienced an event that had an impact on him like no other. At a local jazz jam session the young musician summoned up the courage to ask if he could sit in and promptly called off the tune “Misty”, the jazz standard by Erroll Garner. His nervousness and inexperience became glaringly obvious when during the course of his solo, he missed going to the bridge of the tune. There he was, playing his favorite (and probably only) lick on an Ebmaj7 chord while the rest of the band had begun the modulation to another key. Something had gone very wrong. Unfortunately, he had forgotten the form of the tune and could not hear the harmonic motion of the chords to recover accordingly. Very quickly, the leader pushed him aside and finished playing the song. Afterwards, the leader’s single and very curt suggestion was that the young musician should go home and “woodshed.”

Like the anecdote above, it's quite common for novice musicians to forget their place in the song, miss the bridge, or be unable to play the tune in the key or at the tempo chosen by the leader. Quite often the latter is used as an effective gauge by veteran musicians to determine the abilities of others. Veteran players will call off a particularly fast tune, e.g. "Giant Steps" or "Cherokee," but in an unconventional key. If the musician can't handle it, you may hear the derogatory term that jazz musicians use for this; "skatin' the changes" (the inability of the improviser to play over the chord progression, choosing inappropriate notes for the prevailing harmony) (Berliner, 1994).

Where can young or novice jazz musicians get the training needed to become more effective improvisers? Research suggests that quite often in music education, even when a jazz band program exists, training in improvisation is not a high priority because of the time and dedication the student needs to commit in order to master the skills required (Jones, 2005; Knox, 1996). Or, it may be because of the music teacher's unfamiliarity with the subject. Sadly, there are also fewer venues like the jam session above for musicians to play in today.

In the early to middle part of the last century, young jazz musicians' opportunities for learning were limited to personal relationships with veteran musicians, access to local establishments that featured live jazz, radio broadcasts, and eventually recordings (Berliner, 1994). While the latter was instrumental in expanding the reach of jazz to a wider, more diverse audience, live jazz and the access to veteran musicians were still generally limited to more urban areas.

This changed in the late 1950's and 60's as jazz began to be considered an art form, no longer simply the popular music played in clubs and brothels. Musicians and jazz groups were

playing concerts in venues formerly reserved for orchestras and Western Art music. One such group, The Modern Jazz Quartet, deliberately eschewed the jazz club, with its noisy drinking and dancing clientele, performing only in concert settings (Kernfeld, 1988). While jazz ceased to be the popular music it was in the 1920's – 40's, it was now considered by many to be a musical art form on the level with European classical music. Today, there are well-established repertory ensembles in many major cities, most notably the Lincoln Center Jazz Orchestra and Columbia College's Chicago Jazz Ensemble.

With the aura of being a serious musical form worthy of study, jazz instruction began to be incorporated into the school music curriculum (Mark, 1978). Soon, many high school and middle school programs developed jazz groups, colleges began offering jazz performance and composition/arranging as a course of study, competitive festivals were organized, and an industry of jazz education method books, audio materials, and computer software followed (Mark, 1987).

Computer-assisted instruction (CAI) or educational multimedia (EM) for music education has been in existence for over 30 years and widely used since the early days of personal computers. From the popular *Music Ace* to the ultra-sophisticated *SmartMusic* accompaniment system, a wide variety of CAI music programs are available, ranging in complexity from simple, single subject software in music basics for elementary school students that replace flash cards, to college-level theory, history, counterpoint, and even conducting curricula (Rumery, 1986).

One premise behind CAI is that when students can work at their own pace, covering material as interests and skills dictate, learning is more complete and retention greater. Many music teachers look to CAI as a way to supplement students' experience in and out of class. CAI

also offers the possibility of independent instruction for students who because of distance are unable to pursue live instruction (Taylor, 2003). While it may not be a total replacement for more traditional modes of instruction, the use of CAI for performance training can provide learning opportunities for students in valuable ways.

Why Create iShed?

The purpose of this dissertation was to build a software-based instructional environment that is developed as user-interface expert Don Norman (2004) suggests; sensitive to design that is not done in isolation but with a regard to the whole system with which the user will interact. This is an approach best accomplished by developers who are experienced in the domain (Rheingold, 1993). Most software engineers are not content specialists and, quite often, content specialists are unfamiliar with the capabilities of the development environment or even able to think in terms of activities and design within the software environment.

As Daignault (1996) observed, much of the CAI titles found in music education tend to focus on drill and skill style activities that in essence are a high-tech version of flash cards. Students are evaluated on providing a singularly correct response or performance. In the domain of performance training, some software developers have attempted to use this same model of assessment. Consider this statement from the SmartMusic product literature, addressing its benefit to the student musician:

“SmartMusic shows students how well they’ve performed. Correct notes appear on the screen in green. Wrong notes and rhythms display in red . . .” (SmartMusic, 2006)

Assessment of success in jazz improvisation requires a more nuanced approach and therefore CAI may not be a complete replacement for more traditional modes of instruction. There are however possibilities for integrating environments for communication amongst students and experts that could provide a way for more meaningful assessment and critique.

The use of CAI for performance training could provide new learning opportunities for the novice student of jazz improvisation. Exploration could be encouraged and supported. The software can be extensible, allowing for the addition of new content as the user's skills develop. A well-designed program should provide a range of inter-related activities and customization options, providing the student with meaningful activities and a set of tools that are easy to manipulate and enjoyable to work with.

The software *iShed* is an application that attempts to put into practice the intersection of content delivery, skill acquisition, and student-centered learning in jazz improvisation education. *iShed* is unique in that it supports situated, activity-based learning for jazz improvisation instruction. As will be discussed in a later chapter, there are no commercial titles currently on the market that fully address the issues novice improvisers face in learning how to master this domain.

Connection to CSEME

The software *iShed* has at its core an underlying principle that reflects the lineage of Northwestern University's Center for the Study of Education and the Musical Experience and its ongoing interest in creativity, musical affect and intelligence, and student-centered learning. Dissertations by alumni that relate to this project have focused on the use of technology,

instrument skill acquisition, the intersection of musical thinking and affect, and student mediated learning.

Both Daignault (1996) and Dorfman (2006) examined the impact of computer technology on music education. Daignault's dissertation addresses computer technology and its use as a tool in composition that facilitates creative musical thinking. Dorfman looked at the efficacy of learning to use technology with the aid of multimedia, experience using technology, and learning styles. The *iShed* software provides student-centered activities that encourage the student to create music, allowing for customization according to skills and interests.

In examining skill acquisition, Haston (2004) focused on the development of aural abilities in beginning instrumental training. Haston's treatment group received instruction that included play by ear and call and response activities. Both are valuable in developing *audiation* skills essential to jazz improvisation. The *iShed* software includes ear-training activities to aid the development of melodic, rhythmic, and harmonic recognition.

Woodford (1994) posited that musical thinking, affect, and reflection are intertwined, neither definitive nor standardized. Musical meaning is not static but a process, something to be reinterpreted as expertise grows and understanding changes. In Fodor (1998), student mediated social and musical interactions of pre-collegiate jazz musicians are the places where learning can take place. *iShed* provides a means for student reflection and critical analysis as well communication with peers and mentors, encouraging dialog amongst a community of learners.

iShed strives to provide an environment where exploration and reflection are encouraged and skill development, essential to success in jazz improvisation, is both a meaningful and creative activity. Webster (2005) writes in *Technology Strategies for Music Education*,

“Learning is seen as more effective when approached as *situated in activity* rather than received passively” (p. 70). The software will be constructionist in nature in that it provides opportunities to create music as part of performance training. Students will have control of their environment as they build skill by experiences chosen based on interest and need.

Summary

Fundamentally, the jazz musician’s ability to communicate through improvisation is built from an understanding of the song and its stylistic rules (Baker, 1980). This understanding comes from:

- the aural ability to recognize the qualities of chord structures and harmonic progressions;
- the ability to recognize significant groupings of the harmonic progression within larger forms;
- a level of technical accomplishment on an instrument;
- an understanding of style and stylistic “rules” of behavior within the jazz song structure;

May (2003) proposed a model of jazz instruction with a curriculum that includes theoretical knowledge, acquisition of idiomatic material, experimentation with melody and rhythm, and manipulation of expressive elements. The software described here builds upon this model, providing an immersive and creative learning environment for jazz improvisation.

As posited in “Activity Theory” (Nardi, 1996), “You are what you do.” The *iShed* software will employ multiple ways of “doing music” as a vehicle to understanding and performing improvised jazz. Activities are shared and compared; performance, composition, and

analysis are integrated; historical information is tied to style, practice and culture. New concepts are experienced and skill acquired in multiple ways: from elemental to large form; from listening and analyzing to playing, creating, and moving kinesthetically to; from practicing to performing and recording.

The activities created in the software reflect a synthesis of methodologies articulated in jazz improvisation textbooks and personal communications with jazz masters. *iShed* attempts to put into practice a belief that music education, particularly jazz improvisation, can be made more dynamic and engaging to the novice player through the use of computer technology.

Chapter Organization

This paper accompanies the submission of the software *iShed*, providing a rationale for and explanation of its development. Chapter one has provided an introduction and theoretical foundation as to why there is a need for software such as *iShed*. Chapter two will review the literature related to jazz improvisation and instruction, Computer Aided Instruction (CAI), and user interface design. The third chapter will address the development process with a focus on activity design considerations, engine development, description of the software environment, and an explanation of its proposed use in practice. Chapter four will draw conclusions from reflections on the development of both software and content structures, offering suggestions for the integration of *iShed* into music performance research.

CHAPTER 2 – REVIEW OF THE LITERATURE

This chapter will examine the literature, laying a foundation for the decisions made in the design of the activities and user interface in *iShed*. Two primary bodies of research were examined; jazz improvisation and Computer Aided Instruction (CAI). Within each domain, there are further sub-sets that include educational issues, effectiveness and in the case of CAI, user interface design. Reviews of current jazz improvisation CAI software titles are included at the end of the chapter.

Research on Jazz Improvisation and Instruction

Jazz Improvisation

Improvisation is quite a common activity in our daily lives. Much of what we do every day, far from being completely scripted, unfolds as we go about our business. When walking down a street you may have a general idea that you want to get from point A to point B but what actually happens in between can vary from one day to the next. In conversation, we engage in dialog that may have a general direction but the particular utterances change depending on the situation (Sawyer, 2000). Improvisation in jazz music is very similar.

Is everyone capable of improvising, able to conceive of a melody and perform in real time? There is research to support the notion that jazz improvisation can be taught (Humphreys, May, and Nelson, 1992). If you can hear a melody in your head, you should be able to improvise to some extent. Gordon (1999) uses the term *audiation* to describe this ability, “Audiation is to music what thinking is to language” (1989, p. 77). While we may all be able to improvise,

Gordon sees the development of our *audiation* skills as preparation or “readiness” to improvise. His tests, the *Harmonic Improvisation Readiness Record* and the *Rhythm Improvisation Readiness Record*, are designed to help educators when working with novices (Gordon, 1989).

McPherson’s study (1994) of high school instrumentalists examined the effectiveness of a researcher-designed test to determine the musicians’ ability to improvise. While not specifically using the term *audiation*, McPherson alludes to one’s ability to conceive of the sound in one’s head. In his conclusions, he reports that while technical proficiency is not a significant factor in improvisational ability in the beginner stages, a musician must work on singing as a way to improve their general skills in what he refers to as an “ability to think in sound” (p. 19).

How do musicians improvise? Johnson-Laird (2002) asks the question, going so far as to call it a “riddle” (p. 417), implying that it is something that can be solved, or at the very least described in an algorithm. At its most fundamental level, the task of the jazz musician is to create a “novel melody” (p. 422) within the given harmonic and rhythmic structures. He equates the task with being a ‘transducer’ in automata theory, stepping through the chord sequence in conjunction with the perception of the accompaniment. This requires the musician to access long-term memory and pass information to working, or short-term memory. Pressing (2001) describes improvisation as a multi-step process in which the performer plays a pattern or motivic idea and then, upon recognizing its value, may further develop the idea or decide to pursue a new one.

The examination of the cognitive thought processes musicians go through when improvising is a very difficult thing to do. Not only is it impossible for wind players, but also having subjects verbalize in real time as they are playing may be awkward. As the literature

seems to reveal, improvising is an act that combines rehearsed patterns with subconsciously constructed ideas in the moment (Barrett and Peplowski, 1998). Conscious and subconscious processes meet and operate together at the same time (Seddon, 2005). Therefore, the moment of creation may not be able to be examined since the performer does not have access to their subconscious processes (Sloboda, 1993).

In jazz, when we improvise in an ensemble setting, there is a given structure that all members follow. In essence, we as a unit are imitating what the composer does, but in real time. Are improvisation and composition similar constructs? Is improvisation just spontaneous composition with the distinguishing factor being the time continuum in which the activity is performed? Sarath (1996) is not convinced that the composer's ability to reflect on their work and return to it on any level is similar to the jazz improviser's frame of reference to the time continuum. Composers have the ability to go back and reflect on their work from an endless perspective, re-working down to the elemental level. They can bring compositional process to a halt while they contemplate and reflect on the smallest component, whether to make an adjustment or leave as is.

Since jazz improvisation generally occurs within the framework of a live performance, there is no stopping the forward motion of the music. While it's possible to revisit the generative idea during another passage of the form, the improviser can never stop and look back (Sawyer, 1992). Indeed, degrees of improvisation fall along a continuum. Most jazz musicians' solos rely to some extent on pre-rehearsed patterns, or "licks", that are stitched together depending on the harmonic progression and stylistic considerations. These *licks* are a recognizable part of each

musician's sound and have a strong relationship to the lineage of great players who have come before (Berliner, 1994).

According to Barrett and Peplowski (1998), jazz great Charles Mingus may have said it best, "You can't improvise with nothin'. You gotta have somethin' " (p.558). Baker (1980) warns the uninitiated musician that improvisation in jazz is not just simply playing whatever you want without any concern for organization or structure. Every type of improvised playing requires the player to follow the *emic* or internal rules, to understand the particular style for its "codes of behavior" so they may develop an appropriate strategy for successful navigation (p. 42). The rules for improvising in the style of J.S. Bach are quite different from those of Charlie Parker.

Perhaps the seminal examination into the thought processes, modes of learning, and interactions amongst jazz musicians is Berliner's *Thinking in Jazz* (1994). Berliner's analysis of interviews with professional musicians reveals that within the jazz community there is a system, although implicit, that serves as an educational environment for the novice player. This 'school' can be found for example, in the jam sessions that musicians would frequent in the off-hours of clubs when the scheduled act had finished for the night.

Berliner also examined at musician's development, suggesting a model that describes three stages of development: 1) imitative; 2) assimilative; and 3) innovative. In the *imitative* stage, a favorite musician influences the novice player. They may copy small melodic fragments or entire phrases and incorporate into their improvisations. They may even attempt to sound and act like their object of imitation, going so far as to purchase the same brand of instrument and

clothing, even taking on idiosyncratic mannerisms. In the assimilative stage, the novice begins to incorporate ideas from a larger pool of musicians. They also begin to develop their own “sound” and style based on these influences. Finally, musicians play on a level that is more than the sum of its parts. To be innovative, improvisation must become an expression of both mind and spirit.

In an article with implications for music education, Kratus (1991) described a multi-level system to explain improvisation as well as provide educators with a developmentally appropriate sequence: Exploration, Process-oriented, Product-oriented, Fluid, Structural, Stylistic, and Personal. Kratus defines the exploratory process as “pre-improvisational,” done in a very loosely structured context that is private and not group supported. This feeds into a more process-oriented stage that is more cohesive, including pattern-based material and some *audiated* patterns. As the player moves to more conscious structural considerations including pulsation, tonality, and meter, they become more aware of their performance as product, leading to the player’s manipulations of the instrument or voice in a relaxed, fluid manner. Technical issues become less of concern thereby allowing more attention to the structure of the improvisation. As they progress, the player begins to develop strategies for constructing their solo, either musical or non-musical, and incorporates learned material in a creative manner. In the upper levels of Krataus’ model, the player begins to improvise skillfully within a given style, finally transcending known styles to develop his or her own.

Researchers have also looked at jazz improvisation for its significance in describing social constructs between people. Sawyer (1992) interviewed professional musicians in an attempt to examine improvisational creativity. He suggests that when improvising, musicians operate within both conscious and non-conscious states as they balance their performance

between themselves and those in the group. Seddon (2005) suggests the term “empathetic creativity” as a way to describe the non-verbal interaction that goes on between musicians during a performance. He further breaks down this mode of communication into three parts, *instruction*, *cooperation*, and *collaboration*. Instruction can take place when one member wants to play a tune that the others do not know. Instead of notating on paper, they may resort to playing on their instrument, requiring the others to listen and repeat. Cooperation is represented in the non-verbal cues given between musicians as a means to support the cohesiveness of the ensemble. This can manifest itself in the physical motions a player may do to cue an entrance or begin the song. Finally, collaboration refers to the creative output entirely through non-verbal musical dialog. For example, improvisers may employ a call and response technique with the other members of the ensemble.

Jazz Improvisation and Education

Others have examined jazz improvisation through an educational lens. Elliot’s dissertation (1983), proposes a theoretical and philosophical foundation for jazz education. Reimer’s philosophy of music education as aesthetic education (1970) provides a fundamental component to his thesis in which Elliot proposes an additional dimension of musical meaning. Elliot calls it the “procesual,” a unique characteristic of jazz improvisation that describes the musical feel and time that are completely intertwined with performance.

May’s (2003) work examined the factors that might contribute to achievement in jazz improvisation. Subjects in the study were given three researcher-created tests that examined the subjects’ knowledge of jazz scales, jazz chords, chord substitutions, and harmonic function in

written form, the ability to recognize intervals, scales, chord quality, and melodic patterns from a recording, and the ability to imitate melodic and rhythmic patterns on their instrument from a recording. May concludes that instead of developing skills sequentially, jazz improvisers must develop a “multiplicity of subskills” in a simultaneous fashion (p. 255).

Baudo (1982) looked at jazz improvisation training and its enhancement of the characteristic traits found in creativity that he defines as musical thinking and judgment when creating melodies. Through a survey instrument given to both music educators and high school students, Baudo reported that ninety-four percent of music educators believe that a jazz curriculum enhances musical creativity and eighty-four percent of both educators and students feel that jazz education should be a part of any music instruction.

Fodor (1998) examined the ways pre-collegiate students interacted within the small jazz groups both socially and musically. Fodor’s descriptive study employed Vigotskii’s (1978) “Zone of Proximal Development” and Berliner’s (1994) three-part model (described previously) as a way to describe the patterns of interaction between members as well as coach and student and looked for “moments of critical change,” within the their playing, indicating student’s “movement from one level of ability to another” (p. 41).

Humphreys’ study (1986) examined “improvised accompanying” amongst college music education students and their ability to harmonize and construct accompaniments from a melody played from a recording. He also measured the effectiveness of some sort of “harmonic audiation” and “performance training” (p.193) program. The results point to two observations. First, the training program used did improve student’s ability to harmonize an accompaniment to recorded melodies. Humphreys feels that it is possible to teach students how to audiate harmony.

Second, Humphreys posits that because students seem to perform better at harmonizing by notation rather than in performance, music programs need to increase student's audiation skills.

Researchers have also examined the breadth and depth of jazz education in secondary schools and post-secondary teacher education programs in North America. Fisher's (1981) survey of jazz education and teacher training in Pennsylvania revealed that most respondents were in favor of the inclusion of jazz pedagogy in the music education curriculum. Elliot (1983) specifically examined the state of jazz education in Canada, finding that while post-secondary educators have a positive attitude towards jazz, they were unsure about its current practice and implementation within the teacher-training curriculum. More recently, there have been several studies carried out that examine jazz education in the United States. All report similar findings; that while jazz is a major component of secondary school programs, preparation for teaching jazz is underrepresented in teacher training curricula (Jones, 2005; Knox, 1996; Mack, 1993; Wiggins, 1997).

Summary

It would seem then that jazz improvisation could be a learned skill, providing the instruction focuses on a broad range of concepts, developed in tandem with each other. Much of the research points to ear-training skills as an essential component of the curricula, working towards the development of hearing melodies in our head, what Gordon calls audiation. The novice also needs to become familiar with the syntax of jazz through the imitation of common melodic patterns. This would include melodic (interval, scale, harmonic quality) and rhythmic training.

Also important to the success of improvisation training is the study of stylistic components found in jazz. This includes deeper understanding of jazz styles and practices. As discussed earlier in this chapter, Baker (1980) believes that successful jazz improvisation is based on an understanding of the rules inherent to the particular style.

Finally, as the research seems to suggest, while educators and students alike believe that jazz instruction can enhance the musical creativity of the individual, improvisation training is not well represented across the music education curricula, particularly in teacher training.

Research on Computer Aided Instruction

Computers in Education

In education, computers have become commonplace in so many facets of the learning environment. Over \$5 billion was spent in 1997 – 98 on educational technology (Hickok, 2003). This same report also revealed that in 1999, 90% of schools reported receiving support for technology with 72% coming from state or federal programs. The computer has become an essential component in the daily teaching activities of the American education system. Relatively obscure twenty years ago, the computer can now be found in just about every elementary and secondary school nationwide. But the computer is experiencing a dual existence--as a blessing to some educators who have found innovative and creative uses for the machine and as a nightmare to others who are confused by the technology or, at the very least, feel a sense of helplessness at the amount of attention and funding that the use of computers receive.

Computer technology has played an increasingly dominant role in all aspects of music including production, distribution, and education. Higgins documented the history of CAI (or as

Higgins labels it, CBI, Computer Based Instruction) in music education, highlighting the late 1960's work by Kuhn and Alvin at Stanford University, Deihk and Radocy (DATE) at Pennsylvania State University and Peters at the University of Illinois (Higgins, 1992). In a 1998 technology survey of Illinois music education programs, Reese (1998) reported that 83% of music teachers have had some computer training with 65% reporting using a computer for some reason related to music teaching. Reese (1994), while noting the role CAI can play, looked past the standard "drill and (s)kill" style of many software programs aimed at the music education market. He identified the use of interactive or hypermedia as a tool for students' exploration that is based on a constructivist view of learning. Technology is best utilized, not as simply a tool in the transfer of facts but as a way to encourage students to be thinkers and problem solvers (Willis and Tucker, 2001).

In a more recent survey of K-12 music teachers from around the United States, Taylor (2003) reported that 82% of the respondents believed that computer technology could be used with many or some types of music instruction and learning in the classroom, and over 93% believed this to be the case for outside the classroom activities. Also reported was that only a small percentage of the respondents, about one third, actually integrated technology at least once per week into teaching and learning activities. Taylor's primary recommendation from these results was to increase training workshops for teachers that stressed the relevancy of CAI to its use in the classroom.

Computer Aided Instruction

Mayer and Chandler (2001) looked at educational multimedia in terms of words (or narration) and pictures (including animation) and the ability to describe how systems (e.g. electric lights, home plumbing, etc.) work. Mayer and Moreno (2002) discuss multimedia in the multiple ways that it can represent various principles where this representation, based on cognitive load theory, can be served by selectively combining text, narration, images and animations, and sound. Cognitive load theory addresses the minimization of extraneous information in learning materials so as to direct learners' attention to only relevant information. Mayer and Moreno conclude that meaningful learning involves selecting relevant pieces of information and organizing them into a coherent mental model.

Mayer's and Sims' research (1994) lead them to develop a cognitive theory of multimedia learning based on the idea that we have separate channels for processing information. In the area of learning, Mayer also seeks to enhance the field of study by using the term "multimodal" in place of "multimedia." Multimodal better describes the process of cognition in terms of multiple modalities that can be presented via computer technology that can draw on a full range of audiovisual resources, ranging from text and data to sound and pictures, and process data in a single integrated delivery system. Moreno and Mayer (2002) also discuss the use of Virtual Reality (VR) in the learning experience and its effect on the student's process of knowledge construction. They describe the levels of environments in terms of levels of immersions. These levels can run from no immersion to low, to high immersive, situated environments.

Edelson (2001) identifies some key issues when designing computer-based learning

environments including student motivation, authentic or contextual environments, and support or scaffolding for learning. His model, Learning for Use (LfU) is a three-step process; 1) motivation; 2) knowledge construction; and 3) knowledge refinement. Motivation according to Edelson, is more specific than what is generally thought of in education. In LfU, motivation is a reflection of the student's understanding of a need for new knowledge or growth in a setting where "the student is already reasonably engaged" (p.358). In designing a learning environment, this motivation should spur student curiosity and "create a demand for knowledge" (p. 360). In knowledge construction, the student assimilates new knowledge structures that are based off of direct experiences and connected with existing knowledge. This can also include communication with others. Finally in knowledge refinement, the student connects new knowledge to "other knowledge" and reinforces it for later use. Refinement can also include reinforcement as a means to strengthen the connections to other knowledge structures.

CAI Efficacy

Researchers have used a variety of methods to examine the effectiveness of CAI both with quantitative and qualitative approaches, including surveys of teachers and students, and quasi-experimental studies that employ adjudicated artifacts. Some studies utilized pre-existing software, others newly developed for the express purposes of measuring effectiveness and attitudes.

Orman's (1998) examination of the effectiveness of multimedia attempts to measure achievement from using, and attitudes towards, multimedia instruction. Based on Reese's article (1994) who identified multimedia and interactive instruction as a means to provide a multi-

channel mode of content delivery, Orman designed an environment that utilizes text, audio, and visual content (still and motion) that while presented in a non-linear fashion, allows the learner to navigate freely throughout the software.

In assessing the efficacy of CAI on rhythmic performance of middle school instrumentalists, Smith's (2002) focus was on one small segment of performance: rhythm. Instead of developing custom software, Smith used a widely published and distributed software package, *Music Ace* by Harmonic Vision (2006), for the basis of his testing. His experiment was designed to see if students who use CAI would perform better in their abilities to read and perform rhythms. Half of his participants (n= 120) were provided with software instruction one half hour per week for eight weeks. The other half used no software in their instruction. While the tasks required of the students were performance based, Smith based some of his work on a cognitive field of study called Field Dependence Independence (FDI) which measures the ability of the subject to take knowledge from one domain and apply it to another. Smith reports that while all of the participants' post-test scores increased; there was no statistical improvement of the experimental group over the control group.

Hagen (2004) also used FDI to examine the effects of cognitive style and methods of instruction in piano training. Two methods of instructional technology were examined including the use of a multimedia program. Hagen's subjects used the software as part of their weekly practice (10-12 minutes only per week). A pre-test/post-test instrument was administered that captured student's sight-playing of a melody via MIDI recording. While not able to make full comparisons of each group because of a low n, Hagen does mention that students who used instructional technology reported that it was helpful in improving their sight-playing abilities.

In examining CAI and student achievement, three recent studies have shown that there is no significant difference in achievement when using CAI as opposed to non-technology based learning. Benson (2002) looked at a variety of types of instructional technology, including MIDI sequences, video, and multimedia. While Benson reports no significant differences between each type of instruction, she does point out that the technology was only used for instructional purposes and not for student practice between class meetings. She suggests the need for future research in this area that would instead focus on student practice. Green (2003) examined the effectiveness of multimedia in guitar instruction to eighth grade students. The students ($n=53$) were given the same instructional material with half receiving traditional face-to-face instruction and half via multimedia. Green's results showed no significant difference in the post-test measurement between the two methods of instruction. In his discussion section, Green does posit that designers of CAI software should create software that can adapt to the needs of individual users.

Sheldon, Reese, and Grashel (1999) examined SmartMusic as an accompaniment device in performance preparation by comparing it with two other methods: no accompaniment and live accompaniment. Data were collected using a post-test assessment tool, each participant playing first as a solo and then with an assigned accompaniment method. They report that the two groups who practiced with some accompaniment rated higher on solo performance than the solo performance group with rhythmic accuracy being the key difference.

User Interface

With the increased use of digital technology, questions arise concerning the design of what is commonly referred to as the *user interface* (UI). It was not until computing power had come down in price that significant attention could be paid to developing a non-textual interface. It was the Xerox PARC system, based on a desktop metaphor with point and click graphical icons, that influenced Apple co-founders' Steve Jobs and Steve Wozniak in their development of the Lisa computer (Sen, 1996). Interface design has been strongly influenced by cognitive psychology research as it shares some similar characteristics including; active learning with a focus on real tasks, a constructivist approach to learning with tools that are adaptable by the user, error correction and scaffolding of learning with multiple modes of expression (Soloway, Hays, and Guzdial, 1994).

Interface design has had multiple labels and acronyms over the past years. Some refer to it as Human Computer Interface (HCI), others have reversed it to Computer Human Interface (CHI), and still others call it Man Machine Interface (MMI). Berg (2000) prefers HCI as it puts the emphasis on the user, which is where the interaction should be focused. Many writers, in addressing the issues of UI design, first ask designers to think about what the user will do. In his interview with Rheingold (1993), Donald Norman discusses how developers must be sensitive to design that is not done in isolation but with a regard to the whole system with which the user will interact. Norman feels that developers “need to ask those questions with the entire working environment in mind, with an eye toward the effects our tools will have on that environment” (p. 7). Najjar (1998) addresses this by considering four factors when designing learning environments: the materials, the learner, the learning task, and the test of learning.

Developing the user interface for CAI in music presents many challenges. Most notably is direct manipulation, or what Berg (2000) identifies as a “model of reality” (p. 360). Certainly, some instruction can be easily provided with standard interface elements such as response buttons for yes or no and transport controls for audio playback, but music instruction may also require direct manipulation of musical symbols for theory and composition, manipulation of instruments for performance training, or simulations that represent an environment where music is found. Simulations and direct manipulations in particular can provide a powerful environment where learning is supported by doing (Edelson, 2001).

Kirshbaum (1986) describes the development of a touch tablet input device developed for melodic dictation. In the rationale for the development of such a system, he posits that a conventional paper-based system does not provide feedback, thereby discouraging experimentation, or “learning by debugging” (p. 14). Pan’s (2001) study, conducted to gauge the effectiveness of the user interface, compared approaches of interactivity within the same software package. Pan examined two ways of interacting with the program *Music Ace*: one using a MIDI keyboard, the other a “virtual” software keyboard that is provided in *Music Ace*. The subjects in Pan’s study consisted of non-major students who were taking fundamentals of music class. Pan did not report a significant difference between MIDI keyboards and mouse pointing and clicking.

Summary

There is little doubt that the presence of technology will only increase in educational institutions over the next few years and the use of CAI will continue to increase in the music education classroom.

Is CAI an effective medium for music learning? There is research on both positive and negative sides of this question. While some of the studies described above point to no significant difference in student learning when comparing CAI to non-CAI use, there are indications that there are benefits to the student. While the above research suggests that there may be no improvement in student performance when using CAI, Sheldon, Reese, and Grashel's (1999) findings suggest that students' performance abilities improve to a greater degree when practicing along with an accompaniment that keeps them cognizant of tempo. Hagen reports that students felt more engaged when using CAI.

What is important though to the development of CAI are the decisions made in designing the user interface. As Berg (2000) and Norman (2004) indicate, creating a good interface requires the developer to be sensitive to the how the user interacts with the software as it relates to the subject content.

Review of Current CAI Titles Relating to Jazz Performance Instruction

In this section, I identify examples of software design in music performance CAI that focus on jazz studies. Based on the research and literature cited above (Edelson, 2001; Mayer & Sims, 1994), it would seem that a computer-based instructional system should be one that: 1.

motivates by eliciting interest or curiosity for the subject; 2. provide tools for skill acquisition through direct experience and situational context and; 3. provides scaffolding for student learning and reflection on their work. Historically, commercial CAI programs in music education have focused on skill acquisition in music theory (Daignault,1996), narrative driven presentations in musicology, or “activity” based exploratory/hands on programs for music creation.

SmartMusic Studio (Mac/Win)

SmartMusic (2005) by MakeMusic Inc., the publishers of *Finale* notation software, has been available in various incarnations for more than ten years. While initially released as a hardware device, the system was re-tooled as a software-based system as personal computer power increased. *SmartMusic* has a library of over thirty thousand accompaniments, covering classical solo literature to jazz standards including a portion of Jamey Abersold’s play-along rhythm section accompaniment titles. *SmartMusic* is now being distributed as a component in band method titles including Hal Leonard’s “Essential Elements” series.

SmartMusic is not a content delivery environment per se as it does not contain instructional material. Its strength is its design as a practice tool, allowing users to load and control accompaniment parameters. *SmartMusic* accompaniments are flexible, allowing for control of playback tuning, key change, and the looping of selected portions for repeated playback. The flexibility of the accompaniment to change tempo based on the performers timing is a key feature, allowing for more expressive performances.

The *SmartMusic* software also provides assessment by identifying wrong pitches and rhythms of the user's recorded performance as well as providing a practice report listing the activities completed and amount of time spent on each task. *SmartMusic* can also record an audio track and encode it as an MP3 file.

Sheddin' the Basics – Jazz Piano (Mac/Win)

Another software package that partially meets the of CAI is Roxmedia's *Sheddin' the Basics – Jazz Piano*, created by the present author in collaboration with jazz pianist Michael Kocour. The content of the software focuses on the development of piano chord voicings and voice-leading when playing jazz standards. *Jazz Piano* provides guided instruction, practice environments for skill acquisition, and the ability to capture student's performance of exercises and songs via MIDI keyboard for comparison to pre-recorded expert performance, reflection, and teacher evaluation.

Sheddin' the Basics – Latin-Jazz (Mac/Win)

The second title in the *Sheddin' the Basics* series, created by the present author in collaboration with percussionist Ruben Alvarez, focuses on the responsibilities of the rhythm section when playing Afro-Cuban style Latin-jazz. The software provides historical background, presentations on Cuban sub-styles, video demonstration of commonly used instruments, and a practice environment in which students can control the playback mix of the instruments associated with the Afro-Cuban style, listen to alternate parts, and control tempo. While the

content is directed towards rhythm section instruments, melodic instruments are provided with chord-scale choices for each style. There are no assessment tools built into *Latin-Jazz*.

PG Music - Jazz Piano and Jazz Guitar Master Class (Win)

Published by the same company that developed the intelligent arranging program *Band in a Box*, *Jazz Piano and Jazz Guitar Master Class* are structured as a series of lessons on playing jazz. Content includes basic chord structures and common chord progressions, playing the Blues, idiomatic elements found in the jazz song forms, and improvisation. Both titles also include audio instruction with diagrams and animation of the piano keyboard and guitar fret board respectively. While both titles are only available for the Windows operating system, the recent conversion of the Apple Macintosh line to Intel-based chips has made it possible to run the software on Apple hardware while running the Windows XP operating system. PG Music reports complete compatibility when running the software under this configuration.

JSS Music – Dick Hyman’s Century of Jazz Piano (Mac/Win)

While the focus of this software is on piano performance, this CD-ROM contains stylistic and historical material that may be of interest to any student of jazz. Hyman covers more than sixty different artists including such luminaries as Bud Powell, Duke Ellington, Bill Evans, and Oscar Peterson. The various jazz styles are tied to their geographic homes: New York, Chicago, New Orleans, and Kansas City. Available in two versions, home and professional, the professional version includes a MIDI file playback environment in which the user can control aspects of the performance including tempo and key. Unfortunately, it appears that JSS Music is

no longer continuing development of this software or providing support or updates. The program will not run on Macintosh computers that cannot boot in System 9 (including Classic 9).

PG Music – Band in a Box, MiBac - Jazz (Mac/Win)

Both *Band in a Box* and *Jazz* are in a class of software called “intelligent arrangers” in that they can build MIDI-based accompaniments (e.g. drums, bass, piano, guitar) based on the entry of chord symbols as you would see in a fake book and the selection of the musical style (e.g. swing, country, latin, etc.) desired. *Band in a Box* is perhaps one of the longest running software titles for jazz improvisation education and is a staple in many school band rooms. *Band in a Box* includes a library of pre-arranged songs as well as add-ons called “Paks” that may be purchased separately. These add-ons extend the basic program with additional features and content including extra pre-arranged songs, libraries of musical style algorithms, and instructional content.

Band in a Box also has a feature worth noting that while not explicitly identified as such, could be considered scaffolding for student learning. For example, when generating a harmony in the style of George Shearing, an explanation is provided of the approach (e.g. closed position chords and guitar or vibraphone lead) as well as suggestions for tempo and accompaniment style. This same feature is found in both the Soloist and Melody Creator as well.

Summary

As the literature seems to reveal, the jazz musician's ability to communicate through improvisation is built from an intimate understanding of the song and its stylistic rules. This understanding comes from:

- the aural ability to recognize the qualities of chord structures and harmonic progressions;
- the ability to recognize significant groupings of the harmonic progression within larger forms;
- a level of technical accomplishment on an instrument;
- an understanding of style and stylistic “rules” of behavior within the jazz song structure;

There does not seem to be any one software title that fully addresses these criteria. Even the Jamey Abersold play-along titles available in *SmartMusic* format are primarily backing tracks that do not provide a way for the novice to choose specific parameters within an exercise. *Band-in-a-Box (BIAB)*, while being incredibly feature-rich, is still primarily an intelligent arrangement tool that requires the user to have previous knowledge about chord progressions and scales to be an effective learning tool. In addition, some of *BIAB*'s features do not seem useful to improvisation training. For example, the Generate and Play a Solo feature claims to be able create a solo in the style of any number of great players, but what learning value is found in an algorithmic realization of a Charlie Parker solo when so many recordings of the master himself are readily available? Finally, in the author's opinion, neither of the above titles have user interface designs that support conceptual understanding of jazz theory.

It seems to the author that useful CAI for jazz improvisation study by novices should include content that is reused and revisited in multiple activities and formats, contain fundamental chord types and progressions that are found throughout the jazz literature, and provides flexibility to the user based on interest and skill level. The use of CAI should provide new learning opportunities for the novice. The program should provide a range of inter-related activities and customization options, providing the student with a set of tools that are easy to manipulate and enjoyable to work with. The user interface should include elements that provide scaffolding for learning. Finally, the software should be extensible, allowing for the addition of new content as the user's skills develop. The following chapter will discuss why *iShed* fills these requirements.

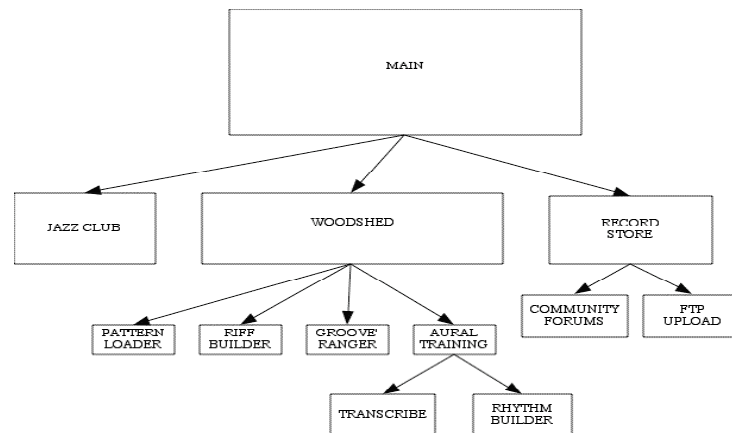
CHAPTER 3 –SOFTWARE DEVELOPMENT

This chapter will describe the development process used to create the software, *iShed*, highlighting its unique features, structure, and the tools utilized in its development. It will address the decisions, problems, and solutions that arose during the design and development stages. First, a description of the software and each activity found within is provided along with a general outline for its use.

Description of the Software

As the flowchart (see Figure 3.1) below indicates, *iShed*'s content is divided into three primary modules: 1) The Jazz Club; 2) The Woodshed; and 3) The Record Store. Within the Woodshed module, activities are further broken down into playing and aural training sub-categories. A description of each activity follows that will describe the user's experience as well as the activity's pedagogical contribution to jazz improvisation training.

Figure 3.1. iShed Flowchart



Software Initialization

When the software first starts, the user is prompted (see Figure 3.2) to test their MIDI set up, select the their instrument's transposition type (i.e. C, Bb, or Eb), and set up their audio recording device. The instrument key selection option facilitates music notation reading by displaying the transposed pitches while sounding the concert pitch equivalent during playback of all exercises and songs.

Figure 3.2. *iShed* setup screens



The Jazz Club

The Jazz Club is considered the “main floor” of *iShed* (see figures 3.3 and 3.4). In this activity the user assumes the role of bandleader, constructing the arrangement and improvising over the entire form of the song.

In the *Jazz Club* the student gains “real word” experience through the arranging and performing of songs. It encourages them to think about the formal elements found in jazz songs beyond the melody including idiomatic introductions, form (e.g. AABA, ABA, etc.) and endings.

Since this simulates a performance venue the goal is to play through a song as if performing for a live audience, without stopping. Live performance requires concentration on the larger issue of form.

There is also a dance-step activity found in the *Jazz Club*. In addition to performing music, clubs traditionally have been places where people dance. There is research that suggests that moving to music promotes kinesthetic awareness within individual styles, tying the jazz literature to dance music and the internalization of rhythmic feel (Downey, 1998; Rohwer, 1997; Crosby 1996).

Figure 3.3. *Jazz Club* arrangement screen

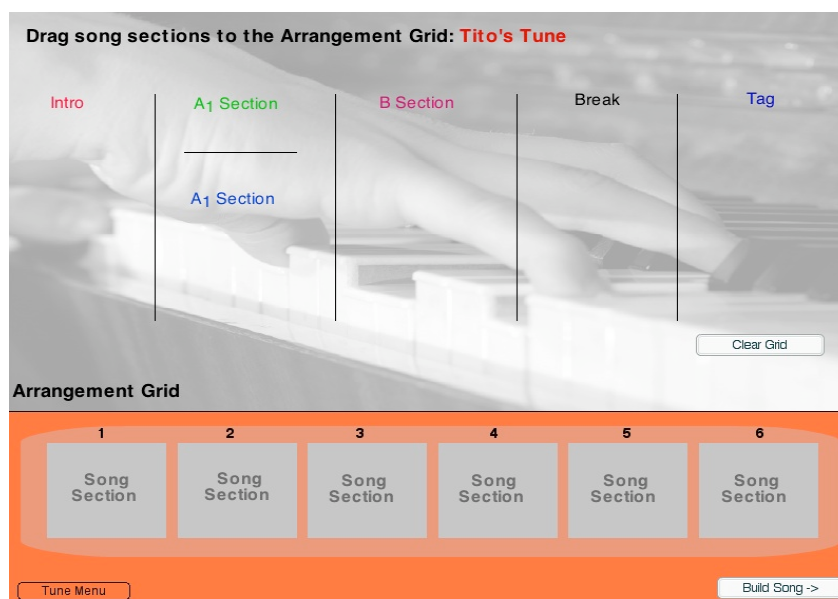
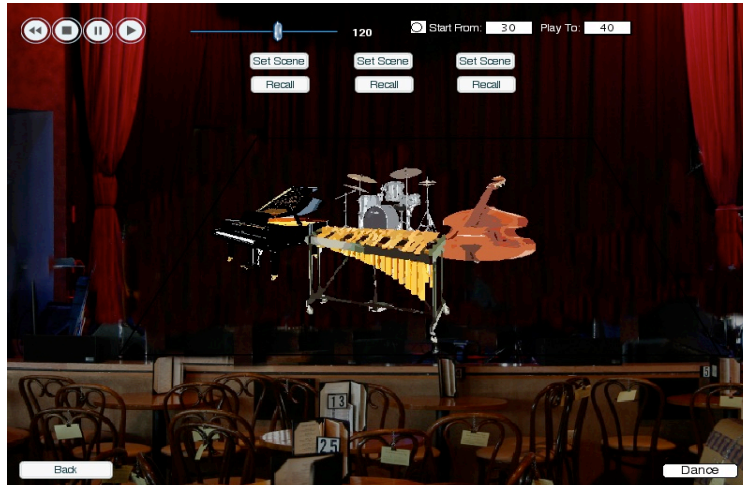


Figure 3.4. *Jazz Club* stage screen



Content

Songs arrangements are constructed by access to a library of eight or sixteen bar phrases that make up the main form of the song and their introductions and endings. Dance steps for the particular style are displayed through animation.

The songs used in *iShed* are original compositions by Michael Kocour, Ruben Alvarez, and the author.

Activity

There are three modes of operation within the *Jazz Club*. The user can act as the arranger, performer, or dancer.

In acting as the arranger, the user must choose the song's introduction, lay out the main sections of the form, and select an ending. This is accomplished by dragging graphics that

represent song components onto an arrangement grid. By rolling their mouse over each graphic, the student will hear the particular section that is associated with the graphic.

After constructing the arrangement, the student is taken to the stage where they can play and record themselves along with arrangement's accompaniment. In addition to the standard transports controls (i.e. play, pause, stop, and rewind), students can adjust tempo and if desired, select a portion of the song for playback. The instruments on the stage act as the controls for adjusting volume and pan. The user can save up to three different configurations, called scenes, of instrument locations.

The students can choose to record themselves on this screen and if they desire, save the recording (.AIF or .WAV) and post via the Message Board found in *The Record Store*.

For each song, there is a button off-stage that will take the user to a screen that presents animated steps to follow and dance along.

The Woodshed

As the flowchart in Figure 3.1 indicated, *The Woodshed* is divided into two sections: Ear Training (*Transcribe* and *RhythmBuilder*) and Playing (*GrooveRanger*, *RiffBuilder*, and *PatternLoader*). The ear training section is further broken down into two activities, *Transcriber* and *RhythmBuilder*. A discussion of each activity found in *The Woodshed* follows.

PatternLoader

The focus of *PatternLoader* is the development of technical skill on an instrument (or singing) by using a vocabulary of motivic patterns. A basic vocabulary of melodic material is

provided and categorized within its tonal placement; major, minor, dominant, diatonic, and harmonic motion (e.g. II-V-I). Students are encouraged to work out the transposition of these patterns in all twelve keys without consulting music notation.

Figure 3.5. PatternLoader activity selection screen

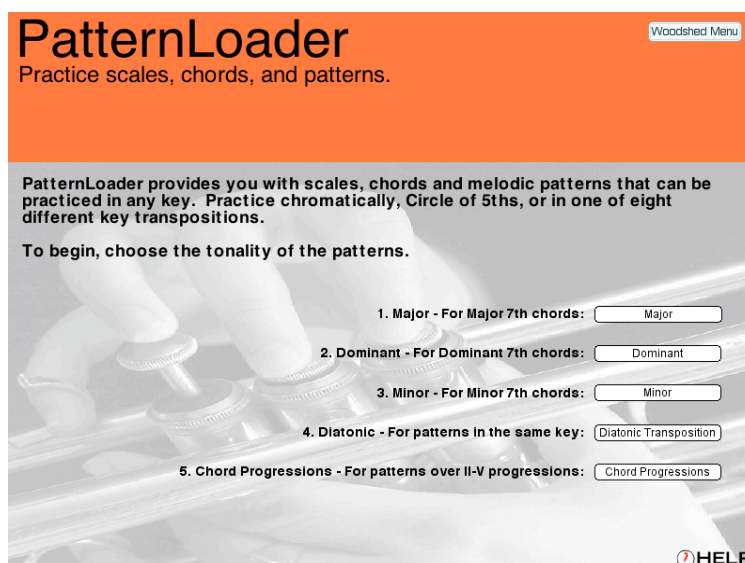
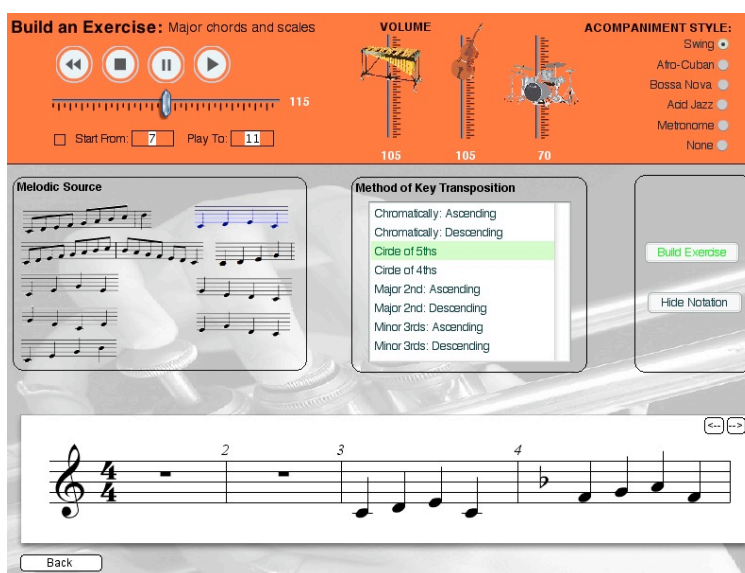


Figure 3.6. PatternLoader activity screen with optional notation displayed



Content

The content in *PatternLoader* includes scales, arpeggios, and one or two bar motivic patterns with the root of the chord. It should be noted that while the patterns presented in *iShed* are very common and could be considered part of the jazz language, some can be found in Jerry Coker's *Patterns for Jazz* (1969).

The percussion accompaniments have been created by the author and include Swing, Afro-latin, Bossa Nova, Acid Jazz, or a simple metronome click.

Activity

The procedure for a typical *PatternLoader* session:

1. Patterns are presented as music notation graphics with an audio cue when the user rolls their mouse over each item.
2. User chooses a pattern, which is then highlighted.
3. User then selects the mode of transposition desired to transpose the selected pattern to the remaining eleven keys. Methods of transposition include ascending and descending chromatically, circle of fifths, circle of fourths, ascending and descending major seconds, and ascending and descending minor thirds.
4. User then clicks a "Build" button, rendering a completed track with the transposition format selected.
5. Percussion accompaniment style defaults to a swing pattern but can be changed at anytime, either before or after track rendering. Transport controls allow the user to play along or record themselves.

6. Volume controls for each track (melodic, bass, percussion) are provided as well as control of tempo.

RiffBuilder

RiffBuilder is similar to *PatternLoader* except that it provides the student with a means to create their own patterns for practice. As with *PatternLoader*, students are encouraged to work out the transposition of these patterns in all twelve keys. *RiffBuilder* employs the “Notation-to-MIDI” engine discussed earlier in this chapter.

Figure 3.7. *RiffBuilder* chord type/progression and key selection screen

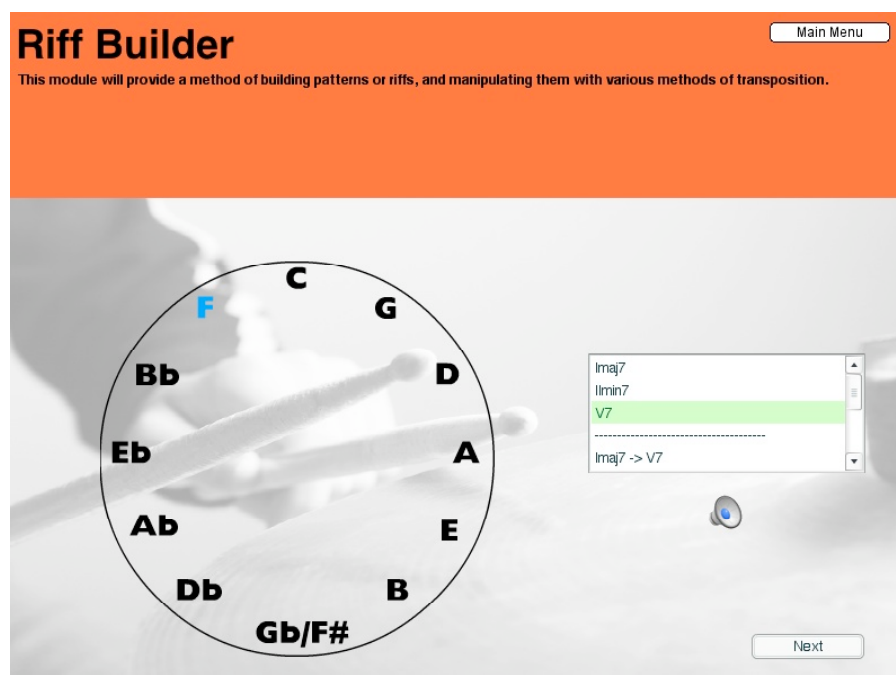
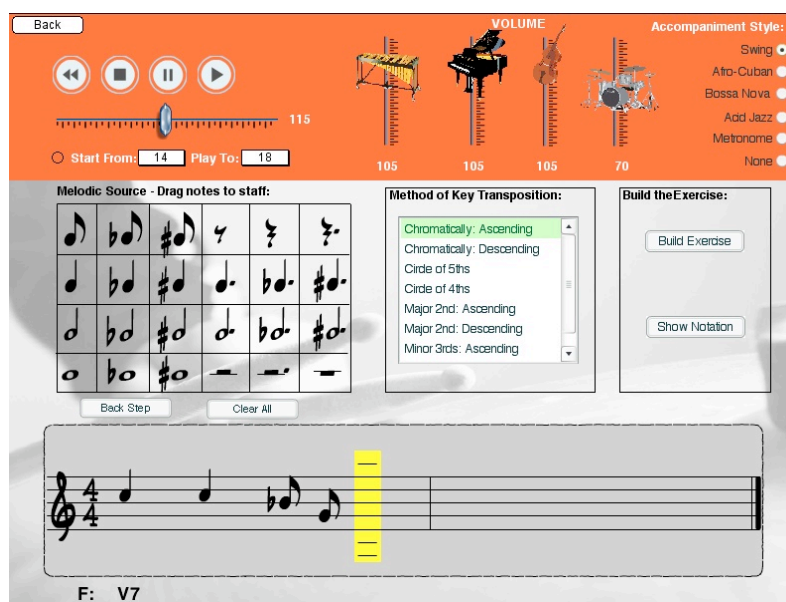


Figure 3.8. *RiffBuilder* notation and transposition screen



Content

The harmonic content for *RiffBuilder* includes chords and chord progressions developed by the author. Individual chord qualities includes major, dominant, and minor seventh chords. Chord progressions include I->V, V->I, I->V->I, II->V, II->V->I, and I->VI->II->V.

Activity

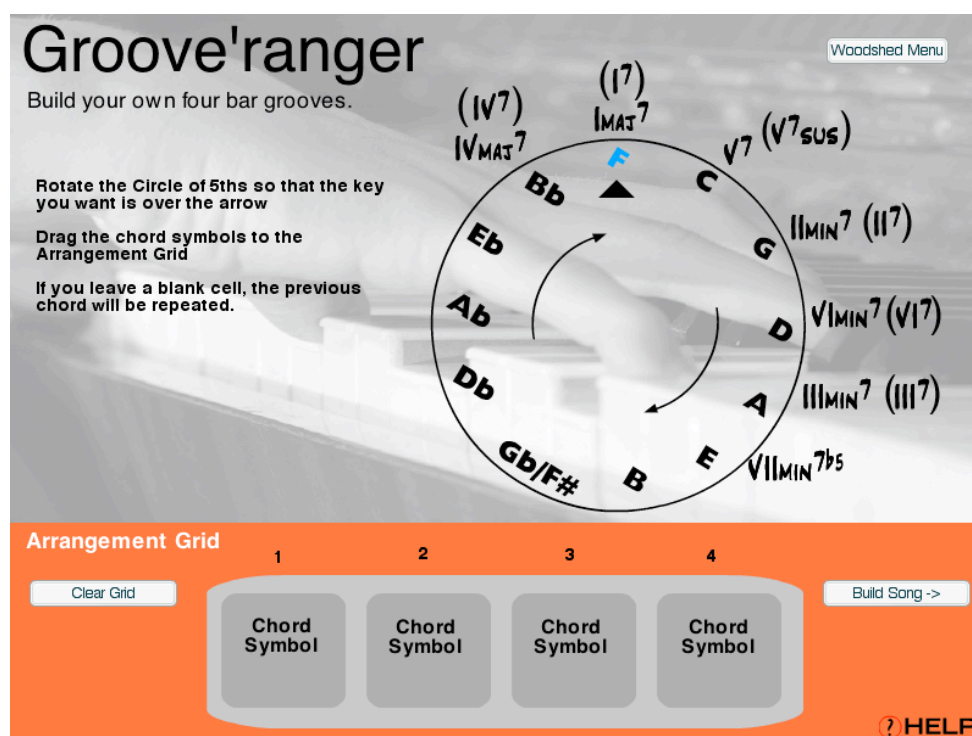
The procedure for a typical *RiffBuilder* session:

1. User chooses a key to be used as the starting point.
2. User chooses a chord or chord progression with which to work from.
3. With the provided palette of note and rest rhythmic values, user drags elements to staff. Feedback includes pitch playback for each note dragged to the staff.
4. User then follows procedures 3 – 6 described above in *PatternLoader*.

Groove'ranger

Groove'ranger provides the user with a means to create short harmonic progressions looped for practice. It employs the “5ths Wheel” described earlier. With it, the user can choose the desired key by clicking and rotating the wheel so that the new tonic of the key lies under the arrow. The user can then drag the chord symbols to the Arrangement Grid.

Figure 3.9. 5ths Wheel w/Chords



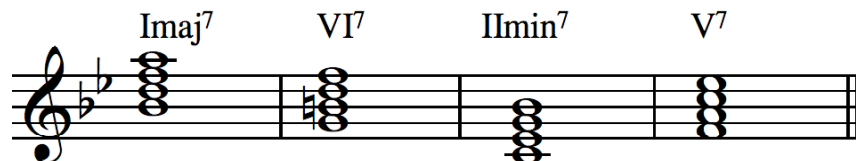
Diatonic chords as well as dominant versions are provided, as it is common to substitute a sub-dominant function chord with its dominant equivalent. For example, the first four chords of George Gershwin's "I've Got Rhythm" are shown in Figure 3.10¹.

Figure 3.10. I-VI-II-V chord progression with minor VI



A common chord substitution is to use a dominant version of the VI chord (see Figure 3.11) that acts as a secondary dominant of II.

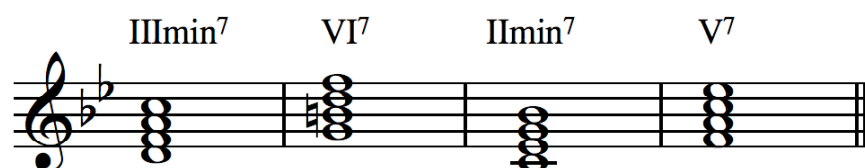
Figure 3.11. I-VI-II-V chord progression with dominant VI



Another common substitution is to use III in place of I (i.e. functional substitution), which then acts as VI7's sub-dominant (see Figure 3.12). Note that with III substituted for I, all of the roots follow the Circle of 5ths.

¹ The chord voicings used throughout this text are in root position to help facilitate faster comprehension of their structure and are not indicative of the voicings used within *iShed*. The voicings, non-chord tones, and voiceleading used within *iShed* are common to jazz piano performance practices. For more information on this topic, please refer to *Sheddin' the Basics – Jazz Piano* by Roxmedia.

Figure 3.12. III-VI-II-V chord progression with dominant VI



Content

The author-created content for this activity is comprised of one-bar MIDI files for each chord quality type. The data files contain piano chord *comping* (i.e. accompaniment) and bass line that is assembled randomly by evaluating which chords are in the arrangement grid.

This activity is extensible, allowing for the addition of new MIDI data files. Those wishing to add additional MIDI files to *Groove'ranger* would place a Standard MIDI File (SMF) for each chord type in the directory called “chordgrooves” that resides in the “midifiles” directory within the “WOODSHED” directory. The SMF file must be configured with the piano part on track one and bass part on track two.

Activity

The procedure for a typical *Groove'ranger* session:

1. User chooses a key to be used as the starting point by setting its position on the “5ths Wheel.”
2. User chooses a chord type and drags it to an arrangement grid.
3. When the user is finished with placing chords onto the grid, they select the “Build” button, rendering the tracks.

4. The user is then presented with a screen representing instruments on a stage similar to the *Jazz Club* discussed above. The playback and recording controls are the same as in the *Jazz Club*.

Ear Training

Transcriber

The *Transcriber* module provides the student with opportunities to develop their aural skills. As shown in Figure 3.20, the melodic material is categorized within its tonal placement; major, minor, dominant, diatonic, and harmonic motion (e.g. II-V-I). There are twenty-one unique examples with each example presented in any of the twelve different keys.

Transcriber is presented in game format (see Figures 3.13 and 3.14). Before game-play, the user selects their skill level that will control the number of opportunities available to check their answer. Each time their answer is evaluated by the software as incorrect, one image of a string bass is removed. Novice users are given four attempts, intermediate users are given three, and experts are given two. After all attempts are used, the correct answer is displayed and a button is displayed that when clicked, displays the next example. A correct response will call a new example.

The *Transcriber* main screen also includes features that will support the student's efforts.

1. "Show Notation" Button – At any point during game-play, the student may select to show the correct answer, displaying it as an overlay on their staff. This overlay is not interactive and requires them to close this window before completing their

response. Clicking the “Show Notation” button does not affect the number of “check answer” opportunities available.

2. “MIDI Keyboard” – The piano keyboard at the bottom of the main screen provides the student with a method to explore their pitch choices before dragging notes to the musical staff.

Figure 3.13. Transcriber content and skill level selection screen

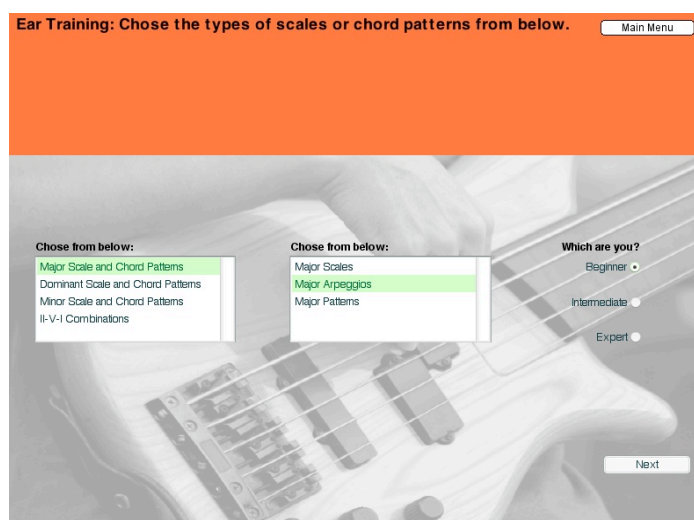
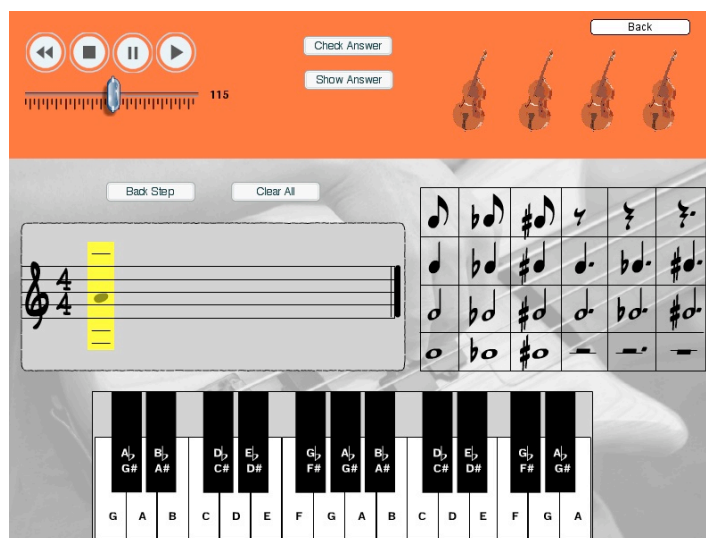


Figure 3.14. Transcribe main game-play screen



Activity

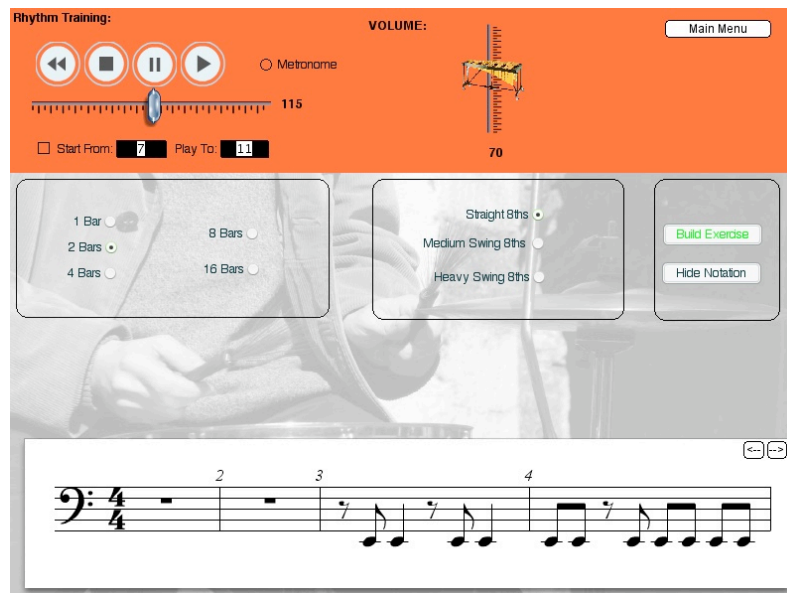
The procedure for a typical *Transcribe* session:

1. User chooses tonality (major, minor, etc.) and melodic type.
2. User chooses level of expertise.
3. User clicks on “Next” button, which moves to the game screen.
4. Upon entry to the game screen, an example is presented aurally with the example’s starting pitch displayed on the music notation staff.
5. User transcribes by dragging notation symbols to the staff.
6. User can play original with or without their notation in the audio mix. Tempo control is also provided.
7. User checks work against “correct” answer by selecting “Check Answer” button.
8. Prompts, both visual and audio are provided for correct/incorrect responses.

RhythmBuilder

The *RhythmBuilder* module provides the student with an interface to build rhythmic patterns of varying lengths from one to sixteen bars. Patterns are constructed with a researcher-created algorithm that randomly selects from twenty-six one-bar source files. Students have the option to either play from memory or read from the optional notation window.

Figure 3.15. *RhythmBuilder* main screen with optional notation displayed.



Activity

The procedures for a typical RhythmBuilder session are as follows:

1. User selects the desired length of rhythmic pattern.
2. User selects the desired rhythmic treatment of eighth note rhythms. Choices include; straight eighths, medium swing, and heavy swing. Medium and heavy swing add a “triplet” feel that is associated with jazz swing to all up-beat rhythmic values.
3. User clicks the “Build Exercise” button.
4. After phrase is built, user can listen to model by clicking “Play” button.
5. With the QWERTY keyboard “space bar” acting as the trigger, user can play pattern.

The Record Store

The *Record Store* provides access to discussion forums, encouraging students to gain knowledge about jazz improvisation by sharing experiences and recordings through use of an Internet forum-based discussion board. Students can also post MP3 formatted audio files that they have created.

The forum, *iShed Message Board*, was created with the on-line service, *Website Toolbox* ([http:// www.websitetoolbox.com](http://www.websitetoolbox.com)). *Website Toolbox* provides application-hosting services including message boards, web site statistics, guestbook, and web site polls.

Design Process

The design process for *iShed* began by identifying and analyzing the various types of tasks that novice improvisers are encouraged to do while practicing. This was accomplished by performing an exhaustive review of the current research literature on jazz education and critically evaluating jazz improvisation method books currently available. As a result of this review, four fundamental task categories were identified: 1) repertoire development through the arranging and playing of tunes; 2) skill acquisition of pattern and motivic repertoire including scales, arpeggios, and idiomatic melodic fragments; 3) melodic/harmonic and rhythmic aural skills development; and 4) an understanding and cognitive processing of chord types and harmonic progressions.

Repertoire Development

Learning the repertoire played by jazz musicians is essential to the musician who wishes to play with others (e.g. sitting in at a jam session). This “canon” of jazz songs, played frequently by jazz musicians all over the world, is referred to as a set of “standards.” In order to avoid the issues associated with gaining permission and paying royalties to copyright holders, the author decided to use original compositions. The chord progressions of the songs are found in many common standard jazz tunes. A variety of song styles were included: swing, Latin-jazz, and acid jazz (a fusion of hip-hop and jazz made popular by such artists as Roy Ayers, the group Medeski, Martin, and Wood, and Liquid Soul).

Pattern/Motivic Skill Acquisition

Similar to spoken language, jazz improvisers must develop a vocabulary of pre-rehearsed motivic patterns that can be used in an almost instantaneous fashion. Berliner (1994) addresses this subject in a chapter entitled, “Getting Your Vocabulary Straight”. He opens the chapter with this observation:

Just as children learn to speak their native language by imitating older competent speakers, so young musicians learn to speak jazz by imitating seasoned improvisers. In part, this involves acquiring a complex vocabulary of conventional phrases and phrase components, which improvisers draw upon in formulating the melody of jazz solo (p. 95).

However, learning a pattern in only one key or modality will not suffice as many jazz tunes contain non-diatonic chords and modulations to other keys. So, in order to have a particular pattern available at any time, the improviser must practice patterns in all keys and modalities. For example, the motif below in Figure 3.16 could be utilized on any chord structure that

contains a major triad. While this example is shown in C major, the jazz improviser should be able to play it in any key (e.g. F, Bb, etc.) as well as transposing it diatonically (modal) within the key of C.

Figure 3.16. Pattern on major-based chord



iShed provides a method of transposition to other keys in eight different ways, representing the most common types of harmonic motion: ascending and descending chromatic, circle of fifths, circle of fourths, ascending and descending major seconds, and ascending and descending minor thirds. The software also provides a method to transpose patterns diatonically within any of the twelve keys.

Aural Skills

As the literature clearly reveals, the development of the musician's ear is essential to becoming a competent jazz improviser (Berliner, 1994; Gordon, 1989; Haston, 2004; Humphreys, 1986; May, 2003; McPherson, 1994;). Whether it's referred to as "audiation," "inner-hearing," or another term, the improviser must recognize chord types and their function within the progression as well as the function of pitches within a melodic line. Berliner (1994) emphasizes this point by recounting an experience a young musician had at a jam session when inquiring about the song's chord progression, " 'Use your ear!' the veteran snapped" (p. 53).

iShed contains two activity areas related to ear training; melodic/harmonic and rhythmic. In the melodic and harmonic activity, scales, arpeggios, and motivic patterns are aurally

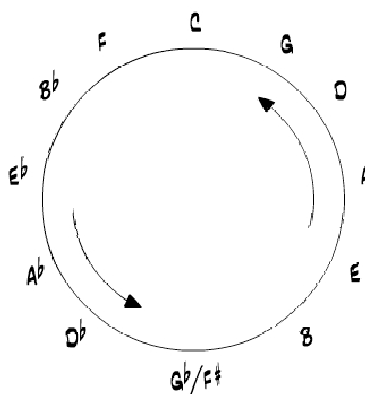
presented randomly in all twelve keys. Users can chose the modality (i.e. major, minor, or dominant) and the type of structure (e.g. scale, arpeggio, etc.). In the rhythm activity, users are able to construct rhythm patterns of varying lengths from one to sixteen bars.

Aural prompts are found throughout the software and students are expected to “use their ears” to learn patterns and key modulations. Music notation is provided but only at the request of the user thereby with the hope that it is accessed sparingly.

Chord Types and Harmonic Progression

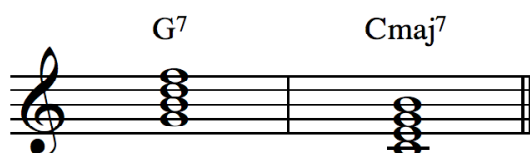
Understanding harmonic progression, the relationship between the chords of a song, is one of the most important concepts jazz improvisers must master, as it is the song’s chord progression that forms the tonal basis of their improvisation. In many methods, a pedagogical approach to understanding harmonic progression involves using a visual representation of key and chord relationships often accomplished with a graphic called the “Circle of 5ths” (see Figure 3.17).

Figure 3.17. Circle of 5ths



The name “Circle of 5ths” refers to the strong tonal motion found in the root motion of a Perfect 5th that is the fundamental motion in Western tonal music, both folkloric and art styles, referred to as the “authentic” cadence (see Figure 3.18).

Figure 3.18. Authentic cadence (V -> I)



Many of the chord progressions found in standard jazz tunes utilize this root motion. Figure 3.19 shows the IImin7 -> V7 -> Imaj7 progression used in the song *Satin Doll*.

Figure 3.19. Cadence progression

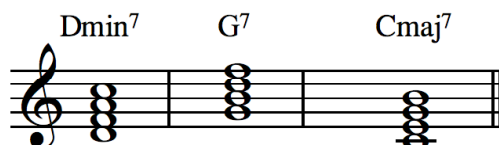
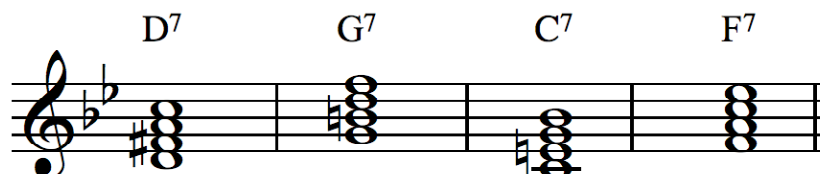


Figure 3.20 is from the bridge of *I Got Rhythm*, III7->VI7->II7->V7. While the qualities of the III and VI chords are not diatonic to the key of Bb (the tonal center of the entire song), the root motion still moves in 5ths.

Figure 3.20. Chords from the bridge of *I Got Rhythm*



Summary

While the above four areas guided the content development of *iShed*, it is important to note that their order presented here is not by importance, nor is one exclusive of the others. The content of *iShed* is not set out in a linear format, requiring the student to master each section before moving on to more advanced concepts. As May (2003) posits, to become an effective jazz improviser requires developing these skills in a simultaneous fashion. And, as Berliner (1994), suggests, improvisational skills are developed in a spiral fashion, with increasing sophistication of student understanding and skill development. The activities found in the software should be revisited as student skill level increases.

Development of iShed

Development Environment

The primary environment used for the development of *iShed* was Adobe's *Director MX 2004*. This program is an excellent tool for multimedia production as it provides quick access to the *application programming interface* (API), supporting fast multi-platform development. Director has been on the market for more than twenty years, first as a product called *VideoWorks*. Director has its own scripting language, *Lingo*, an object-oriented programming language similar to *Java*. Beyond *Lingo*, *Director* supports the development of external classes, called Xtras that extend the functionality of *Lingo* beyond its default classes. The use of Xtras will be discussed below.

A significant portion of the user interface was developed using Adobe's *Flash Professional 8*. Because graphics created in *Flash* are vector images and the rendered files (.swf)

are small in file size, it has become a popular tool of the web development community. *Flash* generated graphics are resizable without the pixilation that can occur when increasing the size of bit-mapped graphics. *Flash* also includes its own programming language, *ActionScript*. Currently in version 2, *ActionScript* is also an object-oriented language. *Flash* could not be used as the primary development environment for *iShed* because *Flash* cannot handle MIDI data. Therefore, using *Flash* movies (.swf) required the passing of variables to *Director*.

Other software tools used in the development of *iShed* included Mark of the Unicorn's *Digital Performer* for the creation of the Standard MIDI Files (.mid), Sibelius, a music notation program by Avid, Inc. for the notation graphics, and Fireworks by Adobe for image editing.

Flash and Director Communication

The main challenge in working with a *Flash* interface running within *Director* is the handling of variables. The data needed to build a MIDI track requires an array of integers that represent timing, pitch, attack velocity, and duration, for each note event. For example, in the Notation to MIDI engine discussed later in this chapter, the solution required passing the data from *Flash* as a text string variable, then parsing back to integers and populating an array within *Director*.

In the interim between programming and writing this paper, this author has found an additional method for passing variables. It is possible to pass integer variables from *Flash* to *Director* by means of the *getURL* method in the *Flash* movie that calls a custom handler in *Director*. Arrays would still need to be repopulated within *Director*.

The method in Flash would look like this:

```
getUrl("lingo:midNote " add noteNum);
```

The handler in Director would look like this:

```
on midNote (noteNum)
    do something with noteNum
end
```

The method used by the author still seems to be the most efficient way to handle *Flash* to *Director* communication. Passing a string that contains all four of the parameters needed requires only one custom handler instead of the four separate handlers the second method would require.

Special Classes (Xtras)

In addition to the Xtras distributed with *Director*, three custom Xtras were used in the development of *iShed*. Each provides a set of classes, properties and methods, which extend the basic functionality of *Director's* programming language.

Sequence Xtra

SequenceXtra, originally developed by the Swedish multimedia development group Sourceforce, is now distributed by Sibelius Ltd. It is a cross-platform Xtra, designed to handle MIDI input and output (I/O) at the port level, import and export Standard MIDI Files and manipulate MIDI data. The current version also includes a notation engine allowing for the display of MIDI data in standard musical notation format but without editing control.

AmiXtra

AmiXtra is a custom cross-platform Xtra developed in consultation with Meliora Software, based in Boston, MA. *AmiXtra* provides methods for recording one digital audio track synchronized with the playback of a MIDI file. It also handles the mixing and playback of Standard MIDI Files (.mid) in conjunction with the digital audio track. Files created with *AmiXtra* are rendered to uncompressed .WAV (Win) or .AIF (Mac) format.

BuddyAPI

The *BuddyAPI*, developed by Gary Smith (Magic Modules) is a widely used Xtra within the *Director* development community. It provides over 140 functions that deal with the Windows API and Macintosh Toolbox. In the context of this project, the *BuddyAPI* was particularly useful for reading the contents of a folder and returning file names. This allows for extensibility of certain activities, thereby allowing users to import new MIDI files with more advanced content.

Core Engine Development

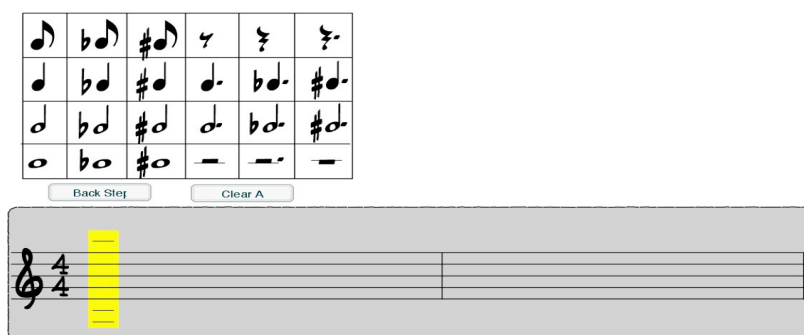
After identifying the four basic tasks addressed earlier in this chapter, the actual software development began by creating the core engines that would be used throughout *iShed*. The engines are a combination of user interface objects and code routines that could be reused depending on the particular need in each of the activities within *iShed*.

1. Notation to MIDI

The “Notation to MIDI” engine is used in the activities *RiffBuilder* and *Transcriber*, providing an interface with which the user can create short melodic passages. The user can drag rhythmic pitch and rest symbols from a palette onto a musical staff for playback

through a MIDI device. Two multi-dimensional arrays are used to keep track of the users' actions, one for the graphical notation elements and one for the MIDI data. To support the user who may not be familiar with the general rules of notation and rhythmic placement within a bar of music, a yellow rectangle highlights the specific location in the measure where the next pitch/rhythm will be placed (see Figure 3.21). Post-placement editing is possible using the “Back” and “Clear All” buttons. Several iterations of this interface were developed and tested by an informal group of users. While there may appear to be redundancy in the notation palette, users reported that this method was the quickest method to move notes and rhythms to the staff compared to the others presented.

Figure 3.21. Notation to MIDI interface



At first, development of this engine was attempted in *Director* using sprite channels to hold media elements. As the work progressed, it became clear that *Director's* capabilities in handling cloned copies of elements from the palette required complex routines to keep track of each element since only one item may be placed in a sprite channel at any given time. After a rough working version had been assembled (taking almost two months), the author decided to abandon it and rebuild in *Flash*, with its better

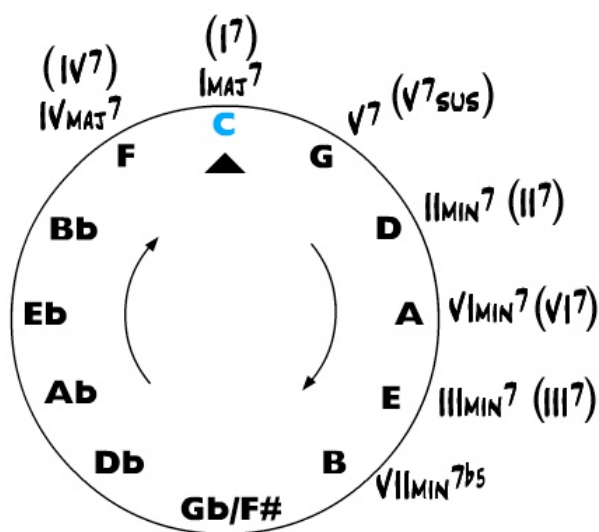
capabilities in handling multiple elements in layers. Building a complete working interface in *Flash* took about half the time as in *Director*.

2. The 5ths Wheel

The “5ths Wheel” is an interactive Circle of 5ths and is one of the very unique elements in *iShed*. With it, the user can easily transpose the MIDI tracks to another key. It also provides a graphical representation of chord types found within a key (see Figure 3.22). The 5ths Wheel is a primary UI component in the activity Groove’ranger (described below) as well as in RiffBuilder and PatternLoader.

The interface was built in *Flash* and has only one variable from the user’s choice, a text string, passed to Director. The use of a conditional statement called a “case” statement is used to determine which key has been chosen and how many half-steps are required for the transposition.

Figure 3.22. Wheel of 5ths from *Groove’ranger*



At first glance, the “Wheel of 5ths” appears to be a relatively simple. But, it became a powerful, pedagogical agent when a rotation behavior was applied to it in *Director*. With the rotation, chord functionality and relationships could be visually represented with a simple rotation of the wheel.

The current iteration of the “Wheel of 5ths” has as serious flaw that has yet to be addressed. The letters do not re-orient themselves to the rotation. When rotating to Gb/F# for example, causes the text to be displayed upside down. This and other flaws in *iShed* will be addressed in Chapter Four.

3. Chunk Arranger

Two activities utilize a method of MIDI track construction where the user builds the song by dragging graphic representations of a chord type or section of a song onto a grid. This is a function found in many MIDI/Audio production environments that use loops for song construction. The Arrangement Grid updates, allowing the user to re-order the elements simply by dragging them from one grid location to another. If the user leaves a grid cell blank, the software will repeat the previous section. A “Clear” button is also provided to reset the entire grid.

4. Parse MIDI data

This is done in every activity found in the software. When *SequenceXtra* returns a method call for the data in a track, it is returned in a multi-dimensional array. One MIDI pitch event is comprised of a five element array (e.g. [2400, 145, 67, 110, 240]):

- i. Position – The start time of an event from the beginning of a song. *SequenceXtra* sub-divides a quarter note into 480 parts per quarter note (PPQN) or *ticks*.
- ii. Event type – Every MIDI event (e.g. note on or off, sustain pedal, volume, etc.) has a unique “type” number.
- iii. Note number – This value can range from 0 to 127, representing a 10 octave range. For example, Middle C on a piano is MIDI note number 60.
- iv. Velocity – The attack velocity is only captured if the users’ MIDI keyboard sends this data type. Otherwise, *SequenceXtra* adds a default value in its place.
- v. Duration - This value is the duration of the particular MIDI event and is also measured in *ticks*.

Routines have been developed that can access and/or modify one or more of these elements simultaneously during playback. For example, when the user chooses a swing style percussion accompaniment, straight eighth note values of the melody must be adjusted to conform to a swinging eighth (triplet-like) rhythmic variant and vice versa. Drum tracks are also constructed without stopping playback (called “on the fly” editing), enabling the user to switch between drum styles (e.g. swing, bossa nova, etc.) with a simple click on a button.

5. Build and transpose MIDI tracks

In many of *iShed's* activities, the user's choice will initiate the loading of a one or two bar MIDI file that form the basis of the specific content. When applying a transposition type to this file, the routine must do two functions. It first copies the data, creating the new measures, and then modifies the note number data to conform to the particular transposition selected. The transposition function was accomplished by creating a multi-dimensional array variable called a "look-up table" that holds the values of each transposition type (see Figure 3.23).

Figure 3.23. Look-up table for major key

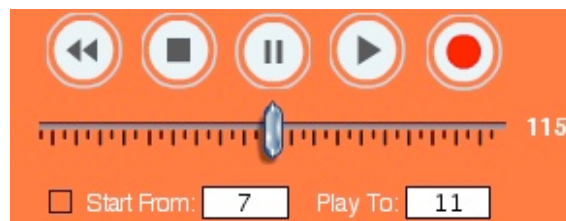
```
majKeySigList = [#chromtc_ascend:[0,-5,2,-3,4,-1,-6,1,-4,3,-2,5],\
#chromtc_descend:[0,5,-2,3,-4,1,-6,-1,4,-3,2,-5],\
#fifths:[0,-1,-2,-3,-4,-5,-6,5,4,3,2,1],\
#fourths:[0,1,2,3,4,5,-6,-5,-4,-3,-2,-1],\
#scnds_ascend:[0,2,4,-6,-4,-2,-5,-3,-1,1,3,5],\
#scnds_descend:[0,-2,-4,-6,4,2,5,3,1,-1,-3,-5],\
#thirds_ascend:[0,-3,-6,3,-5,4,1,-2,2,-1,-4,5],\
#thirds_descend:[0,3,-6,-3,5,-4,-1,2,-2,1,4,-5]]
```

One challenge in the implementation of the transposition function was in the development of an algorithm to display music notation and take into account the need to globally transpose all MIDI pitch events based on the users' selection of instrument transposition. When booting the application, the user is prompted to select the transposition key of their instrument (e.g. Bb trumpet, Eb alto saxophone, etc.) so that when viewing notation, the written pitch and sounding pitch are appropriate to the instrument. The algorithm developed passes a modified note number list to the notation component based on the instrument's transposition.

6. Transport Controls

Transport controls (see Figure 3.24) and their functionality have become quite common in music software. To conform with standard practices, the “Pause” button stops and holds at the current location, the “Stop” button stops playback and rewinds to the beginning of the song. “Rewind” simply restarts at the beginning of the song, responding to the current state of playback, either playing or paused. Tempo control is provided in the form of a horizontal slider with numeric output.

Figure 3.24. Transport controls



7. Mixing Controls

Mixing controls vary in style in the software depending on the environment of the particular activity. Some simply use a slider similar to ones found on mixing consoles. In order to create a more immersive environment in the *Jazz Club* and *Groove'ranger*, a stage was created with direct manipulation of each instrument image. Control of volume and pan (placement in the stereo field) is then accomplished by the relative position of each instrument. To make an instrument louder, the user drags it closer to the front of the

stage, and to the back for soft. Moving each instrument left or right on the stage controls pan.

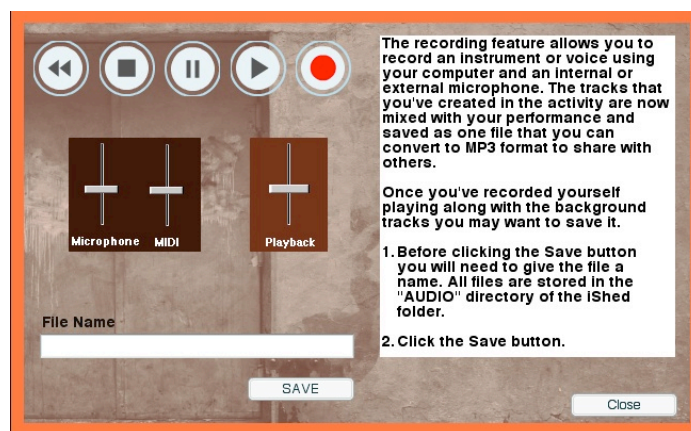
8. Audio Recording with MIDI

One of the primary goals for *iShed* was to provide a way for the students to record themselves while playing along with the MIDI tracks. This required the development of a method of digital audio recording in conjunction with the playback of a MIDI file. This is a feature found in both SmartMusic (Mac/Win) and Band-in-a-Box (only Windows).

With this feature students can record themselves while playing an exercise or song, saving it for reflection and sharing with a mentor or other musicians.

This feature requires providing an interface (see Figure 3.25) for recording that is independent of the current activity's transport controls. When recording an audio track, the MIDI file is also being rendered as digital audio and mixed with the recorded track. At that point it ceases to be able to be controlled by the methods in *Sequence Xtra* and the *AmiXtra* now takes over. The solution was to create a new screen and display only those controls that work with the digital audio recording functions.

Figure 3.25. Transport controls for recording



Summary

Developing the user interface for CAI in music performance training presented many challenges most notably with direct manipulation, or what Berg (2000) identifies as a “model of reality” (p. 360). Certainly, some instruction can be provided with standard interface elements such as response buttons for yes or no and transport controls for audio playback, but music instruction may also require direct manipulation of musical symbols for theory and composition, manipulation of instruments for performance training, or simulations that represent an environment where music is found. Simulations and direct manipulations in particular can provide a powerful environment with which learning is active, engaging process.

CHAPTER 4 – OPPORTUNITIES, CHALLENGES, AND CONCLUSIONS

Innovation and Enhanced Capabilities of Interactive Multimedia

The software, *iShed*, is an application that attempts to deliver content and facilitate skill acquisition in jazz improvisation education. Activities and user interface elements were designed simultaneously, informed by the nature of the domain and the content found within.

How does it surpass currently available titles?

iShed is unique in that there is nothing currently available on the market that integrates interactive content with user-centered activities that promote skill acquisition. As identified in the review of current titles in Chapter 2, the only programs that closely resemble this type of learning environment is *SmartMusic*'s inclusion of the Jamey Abersold play-along accompaniment series into their library and PG Music's *Band-in-a-Box*. As the original Abersold series was only available as audio recordings, this does provide a new level of control over the accompaniment tracks. PG Music's *Jazz Piano Masterclass*, *Jazz Guitar Masterclass*, and Roxmedia's *Sheddin' the Basics Jazz Piano* and *Latin-Jazz* all provide a comprehensive curriculum and interactive set of tools for studying jazz piano, guitar, and Latin-jazz styles respectively but with little attention paid to the tools for learning and practicing melodic improvisation.

What impact will iShed have on music performance training?

iShed's anticipated impact on music education will be one of a model for the further creation of computer-based performance instruction environments. As computing power

increases and development environments like *Flash* and *Director* become even more user-friendly, it will be easier for content specialists to be involved in the process of creating more dynamic and engaging activities.

The engines (algorithms and UI components) developed in *iShed* could be used in the training of other musical genres such as bluegrass, rock and roll, klezmer, Indian Classical (since MIDI can deal with issues of micro-tuning), or even Baroque-style improvisation training that are outside of the jazz domain yet have quasi-improvisational structures inherent in them. For example, in Indian Classical music, while the *ragas* (melodic material or gestures) are fixed elements, the player is given latitude in the combination of *ragas* during performance (Behague, 1984).

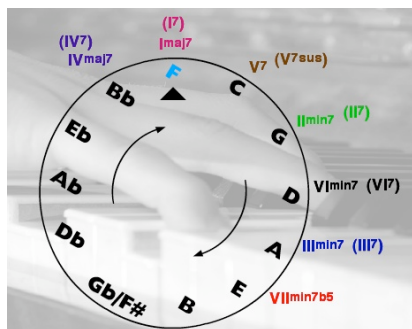
What are the strengths of the software?

iShed's strengths are its user-centered activities that are situated in the types of skills that a jazz player must master to become an effective improviser. Content and practice environments are highly integrated, providing the student with control over the software that is based on interest and skill level. This interactivity cannot be captured and instantiated in any other media format. For example -

- The “5ths Wheel” used in the *Groove'ranger* module provides a highly interactive method for students to assemble chord progression accompaniments. By visually tying the “Circle of 5ths” to chord functionality, students are provided an opportunity to make the association between chord function and key by both visual and aural representations. For example, when using this feature to learn the

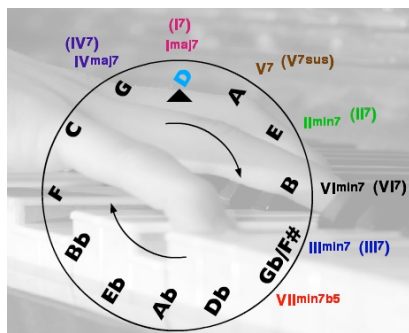
common chord progression called a “II-V”, a student could examine its occurrence in multiple keys. Figures 4.1 and 4.2 demonstrate this capability.

Figure 4.1. *Groove'ranger* “5ths Wheel” set to F major.



- With the “5ths Wheel” set to F major, the student is provided with a visual and audio representation of the II – V progression, clearly showing the II and V chords starting on G and C respectively.

Figure 4.2. *Groove'ranger* “5ths Wheel” set to D major.



- *RiffBuilder* provides the student with tools to create and practice their own patterns or a riff they’ve transcribed from a recording. With this, they can build their own library of patterns, moving from what Berliner (1994) refers to as the imitative stage (found in the activity *PatternLoader*) to the assimilative stage.
- In the *Jazz Club*, students are able to assume the role as the leader of a group by making decisions regarding the arrangement of the tune. They can also

experiment with instrument mixing (i.e. volume and pan) by adjusting the instruments' position on the virtual stage.

- The ability to record an acoustic instrumental or vocal performance as digital audio in conjunction with MIDI tracks, then rendering the two mixed as a single audio file, offers a valuable opportunity for post-practice session analysis to evaluate the student's performance by both peers and mentors, to realize areas requiring further practice for successful execution of technique, and to plan further work on specific skills.

What about assessment?

As addressed earlier, the most difficult function to implement in CAI jazz improvisation performance training may be in the area of assessment. Authentic assessment of success in jazz improvisation surely requires a more nuanced approach than simply identifying incorrect note and rhythm choices. Thom (2006), who is active in the field of Artificial Intelligence and real-time improvisational systems states;

...because artistic quality is ultimately based upon subjective opinions, there is not the same type of idealized success measure that one gets in engineering (compare measuring "Is this compelling?" with "How many zip codes were sorted correctly?") (p. 2)

As jazz artists such as Ornette Coleman and John Scofield have demonstrated, all twelve pitches can be valid choices on any chord, depending upon context. It is found in the abilities of the musician, how they approach and leave a particular pitch cluster that will determine whether a note is perceived as a valid choice or not. Strong rhythmic articulation and attention to stylistic concerns can also factor into the validity of a melodic line (Berliner, 1994).

While it is currently possible for software systems to gauge the correctness of pitch and rhythm as compared against an ideal performance (*SmartMusic* does this very well), assessing improvisation in the way that Thom (2006) identifies as “compelling” may be some years away. Instead, *iShed* relies on self, peer, and mentor assessment. The verbal and musical dialog that takes place between musicians, expert and novice is where learning and understanding can take place. This is what Fodor (1998) and Berliner (1994) point to when examining the dialog that jazz musicians have with each other. Berliner suggests that the young improviser must learn to understand and handle criticism from others, particularly in the context of personal taste, so as to “develop competency in the fundamental language of jazz and to assert their individuality with a world of criticism” (p. 284).

What “issues” remain to be addressed in the software?

There are flaws in the software; some of which were the result of the compressed nature of the development time, some because of programming skills, or lack thereof. If the author has any hope of developing *iShed* into a more mature package, these will have to be addressed.

1. Tracks created in *Jazz Club* song arranging and *Groove'ranger* chord arranging are built from MIDI file chunks. The source files for these routines are fixed structures, assembled using random number generators. Using random selections results, at times, in “un-musical” combinations. Further work must be done to create a more intelligent “arranger” algorithm.
2. There are elements of the UI that, while functional, are rudimentary. For example:
 - a. The circle interface element for the “5ths Wheel” used in *Groove'ranger* is a *Flash* SWF with a behavior attached in Director and does not re-orient

the text. Switching to Gb for example renders the text upside down for the entire circle. This interface needs to be rebuilt entirely in *Flash* to better control text orientation.

- b. The virtual keyboard in the *Transcriber* module only plays the selected pitch via MIDI and does not interface with the note palette and staff. Results from informal user testing suggest that the keyboard should act as a method of note entry on the staff.
 - c. The notation interface used in *RiffBuilder* and *Transcriber* is not fully integrated with the MIDI data. Graphical and MIDI data are separate structures. This prevents editing directly on the staff or a MIDI file from being loaded back into the edit screen. This was a conscious decision made at the start of development that now appears to have many shortcomings. With a longer development time-line (or additional programming assistance), a fully duplexed system could be created that would follow the norms associated with music notation software.
 - d. The dance-step movement activity in the *Jazz Club* is in this developer's opinion, crude at best. The animation needs to be completely reworked to better represent the motion of the feet. The addition of a video demonstration aid would also help convey the total movement of the body.
3. In the original proposal for this project, the author described an environment that was immersive and situated in a "world of jazz." Yet, in planning the development schedule for *iShed*, graphics, animation, and what developers call

“eye candy” were left until the very end of the development cycle and therefore not implemented in the way originally described. Some graphical elements such as buttons and controls are original creations yet others are from the libraries distributed with *Flash* and *Director*. Background images were purchased from the on-line library *istockphoto.com*.

Recommendations for Research

While *iShed* has the potential to provide researchers with a tool with which to examine a number of domains (e.g. attitudes towards the use of technology or gender and technology) the fundamental question that should be generated by this project is whether the software is an effective tool in jazz improvisation education. For example, a major emphasis of the software focuses on the development of skills in playing scales, chord types, and other idiomatic patterns, played in all keys using various transposition patterns (see Ch. 3, *PatternLoader*). A quasi-experimental study could use these patterns as benchmarks, testing an experimental group who use the software as a practice aid with a control group who uses more traditional practice aids. An extension of this study would be to look at the acquisition of those skills as a predictor of the user to play a more-coherent improvisation. Evaluation of improvisation could incorporate the use of a panel of judges who employ May’s (2003) Instrumental Jazz Improvisation Evaluation Measure.

Sarath (2002) advocates that improvisation be studied from a “multi-stylistic perspective” transcending the assumption that improvisation only occurs in jazz (p. 195). Could *iShed* engines provide the foundation for improvisation studies in other styles of music? This may be of

particular interest to music educators and teacher-training institutions, as indicated in Chapter 2, the literature seems to reveal a general lack of understanding amongst music educators about improvisation, jazz or otherwise.

Conclusion

iShed is the culmination at this point in time of this author's continuing investigation into the intersection of jazz instruction and learning through technology. It is an attempt to build a software-based learning and practice environment that is informed by both research into jazz improvisation education and Computer Aided Instruction. It puts into practice a personal philosophy that is geared toward learner-mediated education, giving students control over their practice environment in ways that are situated within the domain of jazz improvisation.

The content presented in *iShed* is informed by the literature found in Chapter Two which seems to suggest that the core activities jazz musicians do when practicing their improvisational skills should include: 1) honing technical skills (i.e. "developing chops"); 2) assimilating new melodic material from listening and analyzing recordings and live performances of other musicians; 3) developing a stronger theoretical understanding and aural recognition of harmonic progressions and chord substitutions; and 4) finding new ways to arrange and improvise over a song's form and chord progression.

What makes *iShed* unique is in its approach to presenting content in an interactive, activity-based environment. By providing a high level of interactivity to support the building of a short chord sequences, the module *Groove'ranger* is one such example. With its *Wheel of 5ths* interface, the user is presented with both visual and aural support while assembling a chord

progression. Visually, the use of Roman numeral chord symbols in conjunction with the ability to reposition the Circle of 5ths in relation to the chord symbols reinforces the concept of chord functionality in relation to the prevailing key center. The user also receives aural support by hearing the chord sounded when they roll the mouse cursor over a symbol. Multiple senses are engaged when seeing the chord symbols presented around the Circle of 5ths and hearing as each chord is placed in the arrangement grid.

If we were somehow able to whisk our young improviser in Chapter One from the humbling experience at the jam session to a practice session today we may see individual players using *iShed* in preparation for their next performance opportunity. They would be using *PatternLoader* to hone their technical skills on scales, arpeggios, and commonly used motifs. *RiffBuilder* would be used to create melodies over basic chords and progressions providing our novice improviser an outlet with which to build their own riffs and licks.

As a player's experience and skill level increase, the same tools will facilitate movement from imitative to assimilative improvisation, as mastery of the basics are developed and new ideas tested and incorporated into the musician's vocabulary (Berliner, 1994). *RiffBuilder* can also serve to hone audiation skills (discussed in Chapter 3), as the user creates melodies over basic chords and progressions and then employs those same progressions as starting points in *Groove'ranger*. Using *iShed* can also aid our novice musician in developing a deeper understanding of jazz improvisation through the connections made between melody, harmony, rhythm, and style.

Finally, because of remote location or other mitigating factors, we may also see our novice musician become involved in a cyber-community of jazz musicians through use of the

iShed Record Store. This will provide an opportunity to interact with other musicians who may be struggling with similar problems or experts ready to offer a critique of posted recordings. In the end, if *iShed* simply creates a desire to practice a little more and explore a bit more, then the present author will feel that a primary objective has been accomplished.

REFERENCES

- Azzara, C. D. (2002). Improvisation. *The New Handbook of Research on Music Teaching and Learning*. Oxford: Oxford University Press.
- Baker, D. (1974). *Advanced Improvisation*. Chicago: Maher Publication.
- Baker, D. (1980). Improvisation: A tool for music learning. *Music Educators Journal*, 66(5), 42-51.
- Band in a Box. (2006). Vancouver, BC: PG Music.
- Barrett, F., & Peplowski, K. (1998). Minimal structures within a song: An analysis of "All of Me". *Organization Science*, 9, 558-560.
- Bash, L. (1982). Jazz education given a third wave perspective. *International Association of Jazz Educators*, 14-18.
- Baudo, J. M. (1982). The effectiveness of jazz education on the enhancement of the characteristic traits associated with music: Implications for curriculum planning. Unpublished Dissertation, State University of New York, Buffalo.
- Behague, G. (Ed.). (1984). *Performance practice : Ethnomusicological perspectives*. Westport, CT: Greenwood Press.
- Benson, C. (2002). The effects of instructional media on group piano student performance and attitude. *Journal of Technolgy in Music Learning*, 1(2), 38-55.
- Berg, G. (2000). Human-computer interaction (HCI) in educational environments: Implications of understanding computers as media. *Journal of Educational Multimedia and Hypermedia*, 9(4), 349-370.

- Berliner, P. (1994). *Thinking in jazz: The infinite art of improvisation*. Chicago: University of Chicago Press.
- Bruner, J. (1960). *The Process of Education*. Cambridge, MA: Harvard University Press.
- Coker, J. (1964). *Improvising Jazz*. New York: Simon & Schuster, Inc.
- Coker, J. (1970). *Patterns for Jazz*. Miami, FL: Studio P/R.
- Crosby, J. F. (1996). *Will the real jazz dance please stand up? A critical examination of the roots and essence of jazz dance with implications for education*. Unpublished Dissertation, Columbia University Teachers College, New York City.
- Daignault, L. (1996). *Children's creative musical thinking with the context of a computer-supported improvisation approach to composition*. Unpublished Dissertation, Northwestern University, Evanston.
- Dorfman, J. (2006). *Learning music with technology: The influence of learning style, prior experiences, and two learning conditions on success with a music technology task*. Unpublished Dissertation, Northwestern University, Evanston.
- Downey, G. J. (1998). *Incorporating Capoeira: Phenomenology of a movement discipline*. Unpublished Dissertation, The University of Chicago, Chicago.
- Edelson, D.C. (1999). Addressing the challenges of inquiry-based learning through technology and curriculum design. *Journal of the Learning Sciences*, 8(3), 391-450.
- Edelson, D. C. (2001). Learning-for-use: A framework for the design of technology-supported inquiry activities. *Journal of Research in Science Teaching*, 38(3), 355-385.

- Elliot, D. J. (1983). *Descriptive, philisophical and practical bases for jazz education: A Canadian perspective*. Unpublished Dissertation, Case Western Reserve University, Cleveland.
- Fisher, L. (1981). *The rational for and development of jazz courses for the college music education curriculums*. Unpublished Dissertation, The Pennsylvania State Univeristy, University Park.
- Fodor, D. B. (1998). *Critical moments of change: A study of the social and musical interactions of precollegiate jazz combos*. Unpublished Dissertation, Northwestern University, Evanston.
- Gordon, E. (1989). Audiation, Music Learning Theory, Music Aptitude, and Creativity. *Suncoast Music Education Forum on Creativity*, 75-81.
- Gordon, E. (1999). All about audiation and music aptitudes. *Music Educators Journal*, 86(2), 41-44.
- Gordon, E., & Dalby, B. (2002). Audiation. Message posted to <http://www.unm.edu/~audiate/intro.htm>
- Green, B. R. (2003). *The comparative effects of computer-mediated interactive instruction and traditional instruction on music achievement in guitar performance*. Unpublished Dissertation, The University Of British Columbia, Vancouver.
- Hagen, S. (2004). The effects of computer assisted instruction and cognitive style on sight playing among university group piano students. *Journal of Research in Music Education*, 2, 44-55.

- Haston, W. A. (2004). *Comparison of a visual and an aural approach to beginning wind instrument instruction*. Unpublished Dissertation, Northwestern University, Evanston.
- Hickok, E. (2003). *Federal funding for educational technology and how it is used in the classroom: A summary of findings from the integrated studies of educational technology*. U.S. Department of Education.
- Higgins (1992). Methodologies in music education. In R. Colwell (Ed.), *The handbook of research on music teaching and learning*. Oxford: Oxford University Press.
- Humphreys, J. (1986). Measurement, prediction, and training of harmonic audiation and performance skills. *Research in Music Education*, 34(3), 192-199.
- Humphreys, J. T., May, W. V., & Nelson, D. J. (1992 In R. Colwell (Ed.), *The handbook of research on music teaching and learning*. Oxford: Oxford University Press.
- Jazz Guitar Masterclass. (2006). Vancouver, BC: PG Music.
- Jazz Piano Masterclass. (2006). Vancouver, BC: PG Music.
- Johnson-Laird, P. N. (2002). How jazz musicians improvise. *Music Perception*, 19, 415-442.
- Jones, H. E. (2005). *Jazz in Oklahoma music teacher education*. Unpublished Doctoral Dissertation, The University of Oklahoma, Norman.
- Kernfeld, B. (Ed.). (1988). *The New Grove dictionary of jazz*. New York Macmillan Press.
- Kirshbaum, T. M. (1986). Using a touch tablet as an effective, low-cost input device in a melodic dictation CAI game. *Journal of Computer Based Instruction*, 13(1), 14-16.
- Knox, D. C. (1996). *Status of jazz education in the preparation of music educators in Alabama colleges and universities* Unpublished Doctoral Dissertation, The University of Alabama, Tuscaloosa.

- Kratus, J. (1991). Growing with improvisation. *Music Educators Journal*, 78(4). 35-40.
- Lavender, P. (Ed.). (2000). *Essential Elements 2000*. Milwaukee, WI: Hal Leonard Publishing.
- Mack, K. D. (1993). *The status of jazz programs in selected secondary schools of Indiana, 1991-1992*. Unpublished Dissertation, Ball State University, Muncie.
- Madura, P. D. (1996). Relationships among vocal jazz improvisation achievement, jazz theory knowledge, imitative ability, musical experience, creativity, and gender. *Journal of Research in Music Education*, 44(3), 252-267.
- Mark, M. L. (1978). *Contemporary music education*. New York. Schirmer.
- Mark, M. L. (1996). The acceptance of jazz in the music education curriculum: A model for interpreting a historical process. *Bulletin of the Council for Research in Music Education*, 92, 15-21.
- May, L. (2003). Factors and abilities influencing achievement in instrumental jazz improvisation. *Journal of Research in Music Education*, 51(3), 245-258.
- Mayer, R. E. (2001). *Multimedia learning*. New York. Cambridge University Press.
- Mayer, R. E., & Chandler, P. (2001). When learning is just a click away: Does simple user interaction foster deeper understanding of multimedia messages? *Journal of Educational Psychology*, 93, 390-397.
- Mayer, R. E., Heiser, J., & Lonn, S. (2001). Cognitive constraints on multimedia learning: When presenting more material results in less understanding. *Journal of Educational Psychology*, 93, 187-198.
- Mayer, R. E., & Moreno, R. (2002). Aids to computer-based multimedia learning. *Learning & Instruction*, 12(1), 107-119.

- Mayer, R., & Sims, V. (1994). For whom is a picture worth a thousand words? Extensions of a dual-coding theory of multimedia learning. *Journal of Educational Psychology*, 83(3), 389-401.
- McPherson, G. (1994). Evaluating improvisational ability of high school instrumentalists. *Council for Research in Music Education*, 119, 11-20.
- Moreno, R., & Mayer, R. E. (2002). Learning science in virtual reality multimedia environments: Role of methods and media. *Journal of Educational Psychology*, 94(3), 598-610.
- Music Ace. (2006). St. Charles, IL: Harmonic Vision.
- Najjar, L. J. (1998). Principles of educational multimedia user interface design. *Human Factors*, 40(2), 311-324.
- Nardi, B. (Ed.). (1996). *Context and Consciousness: Activity Theory and Human-Computer Interaction*. Cambridge: MIT Press.
- Norman, D. A. (2004). Introduction to this special section on beauty, goodness, and usability. *Human-Computer Interaction*, 19, 311-318.
- Orman, E. K. (1998). Effect of interactive multimedia computing on young saxophonists' achievement and attitude. *Journal of Research in Music Education*, 46(1), 62-74.
- Pan, K. C. (2001). *The effect of computer-based music education software with MIDI versus virtual keyboard use on student achievement and opinion*. Unpublished Dissertation, The University of Alabama, Tuscaloosa.

- Peters, G. D. (1974). *Feasibility of computer-assisted instruction for instrumental music education*. Unpublished Dissertation, University of Illinois, Champaign-Urbana.
- Pressing, J. (2001). Improvisation: Methods and models. In J. A. Sloboda (Ed.), *Generative processes in music: The psychology of performance, improvisation, and composition* (pp. 129-178). New York: Oxford University Press
- Reese, S. (1994). Music technology: Tools for extending and sharing minds. *The American Music Teacher*, 43(6), 12-13.
- Reese, S. (1998). Music learning in your school computer lab. *Music Educators Journal*, 85(3), 31-36.
- Reimer, B. (1970). *A philosophy of music education*. Englewood Cliffs: Prentice-Hall.
- Rheingold, H. (1993). *The Virtual Community*. Reading, PA: Addison-Wesley.
- Rohwer, D. A. (1997). *The effect of movement instruction on sixth-grade beginning instrumental music students' perception, synchronization, and performance with a steady beat*. Unpublished Dissertation, The Ohio State University.
- Rumery, K. R. (1986). Computer applications in music education. *Technological Horizons in Education*, 14(2), 97-99.
- Russell, G. (1953). *The Lydian chromatic concept of tonal organization volume 1 : The art and science of tonal gravity*. Cambridge, MA: Concept Pub. Co.
- Sarath, E. (1996). A new look at improvisation. *Journal of Music Theory*, 40(1), 1-38.
- Sawyer, R. K. (1992). Improvisational creativity: An analysis of jazz performance. *Creativity Research Journal*. 5. 253-263.

- Sawyer, R. K. (1999). Improvised conversations: Music, collaboration, and development. *Psychology Today*, 27(2), 192-205.
- Sawyer, R. K. (2000). Improvisational cultures: collaborative emergence and creativity in improvisation. *Mind, Culture, And Activity*, 7(3), 180-185.
- Seddon, F. A. (2005). Modes of communication during jazz improvisation. *British Journal of Music Education*, 22, 47-61.
- Sen, P. (1996). *Triumph of the nerds*. [Video]. USA: RM Associates.
- Sheddin' the Basics - Jazz Piano. (2000). West Chester, PA: Roxmedia.
- Sheddin' the Basics – Latin-jazz. (2002). West Chester, PA: Roxmedia.
- Sheldon, D. A., Reese, S., & Grashel, J. (1999). The effects of live accompaniment, intelligent digital accompaniment, and no accompaniment on musicians' performance quality. *Journal of Research in Music Education*, 47(3), 251-265.
- Sloboda, J. (1993). Musical ability. In G.R. Bock & K. Ackrill (Eds.). *Ciba Foundation Symposium 178: The origins and development of high ability*. Chichester, UK: Wiley DePublisher.
- SmartMusic Studio (Version 8.5). (2005). Eden Prairie, MN: MakeMusic.
- SmartMusic & Your Students. (2006).
- http://www.smartmusic.com/teachers/default.aspx?page=page1_1_1
- Smith, K. (2002). *The effectiveness fo computer-asssisted instruction on the development of rhythm reading skills among middle school instrumental students*. Unpublished Doctoral Dissertation, University of Illinois, Urbana.
- Soloway, E., Hays, K., & Guzdial, M. (1994). Learner-centered design. *Interactions*(April).

- Taylor, J. (2003). The status of technology in k-12 music education. *Journal of Technology in Music Learning*, 2(2), 48-56.
- Thom, B. (2006). *Research Statement*. <http://www.cs.cmu.edu/~bthom/ME/research.pdf>.
- Vygotskii, L. (1978). *Mind in Society: The development of higher psychological processes* (M. Cole, Ed.). Cambridge, MA: Harvard University Press.
- Walls, K. C. (1997). Music performance and learning: The impact of digital technology. *Psychomusicology*. Vol, 16(1-2), 68-76.
- Webster, P. (2002). Computer-based technology and music teaching and learning. In R. Colwell & C. Richardson (Eds.), *The New Handbook of Research on Music Teaching and Learning*. New York: Oxford University Press.
- Webster, P. (2005). Creative thinking and music technology. In F. Richmond (Ed.), *Technology Strategies for Music Education*. Wyncote, PA: Technology Institute for Music Educators.
- Wiggins, I. (1997). *An assessment of the status of jazz music education in the public high schools of North Carolina: A foundation for curriculum revision in higher education*. Unpublished Dissertation, The University of North Carolina at Greensboro.
- Willis, E. M., & Tucker, G. R. (2001). Using constructionism to teach constructivism: Modeling hands-on technology integration in a preservice teacher technology course. *Journal of Computing in Teacher Education*, 17(2), 4-7.
- Woodford, P. G. (1994). *Development of a theory of transfer in musical thinking and learning based on John Dewey's conception of reflective thinking*. Unpublished Dissertation, Northwestern University, Evanston.