the help of a number of online resources. You can always peruse the code of the Web page you are viewing. With Netscape, click on the View menu and Page Source or type Ctrl + U. With Internet Explorer, click on the View menu and Source.

To create dynamic and interactive Web pages, add JavaScript to your html documents. Begin by inserting free code that is available online into your html files. Next, create your own code. Once again, online assistance abounds. If you are very ambitious, explore Java and Flash.

For additional information, go to www.northern.edu/wieland and click on Why create Web pages?

SEVENTH ANNUAL NATIONAL SYMPOSIUM ON MUSIC INSTRUCTION TECHNOLOGY (2005)

Lessons Taught and Lessons Learned: What My Students Have Taught Me about Teaching Music in a Technology Lab

Alex Ruthman,
Cranbook Schools and Oakland University

For the past two years, I have been trying to find ways to engage meaningfully my middle school students in creating original music mediated by technology. I have tried many things; a few of them have failed, yet many projects exceeded both my and my students' expectations. This presentation shares some of our successful composing experiences, the tools that helped make them work, and insights into what contributed toward musically engaging and educationally appropriate projects.

Lessons Taught

Three approaches to structuring composing experiences will be shared and discussed. Starting with open-ended exploratory composing experiences, I will share how we used two loop-based creative music programs for the PC: Super Duper Music Looper and Acid Music Studio 5.0. Using these tools during introductory class experiences provided a useful window into students' prior musical understanding, illuminating ways we as teachers may better connect to our students' prior musical experience to design appropriate follow-up composition projects. Throughout this presentation I will share and discuss examples of student work, including how unique aspects of the technologies enabled assessment of musical growth and understanding.

Free Composing

Each semester our *Composers' Workshop* class begins with free, open-ended composing experiences that allow students to explore and experiment with the sound possibilities afforded by loop-based music software. Drawing on pedagogical structures developed from writing workshop and "whole language" approaches to teaching writing (Atwell, 1998; Calkins, 1994), instruction occurs in the form of "mini-lessons" at the beginning or end of class, supported by teacher and peer "conferring" throughout classes as compositions are formed and developed. As students encounter problems (musical and technical) support is given emergently, tailored to the needs of the student. Peer collaboration is encouraged throughout the process of composing as students complete a variety of collaborative "digital arts" projects (similar to those designed by Savage, 2005; Savage & Challis, 2002).

The "Rite of Spring" Project

The *Rite of Spring* Project was the result of a curricular collaboration with a middle school creative dance class[1] at my school. In this project, our students collaborated to create an original interpretation of Stravinsky's *Rite of Spring* using loop-based creative music software. Students in my class listened to and analyzed the music of the *Rite of Spring*, creating musical architecture maps of selected movements that served as guides for creating their own original version. Recordings of the *Rite of Spring* were continually listened to throughout the entire project and seemed to be a major enabling factor of the students' success in the project. When students had finished composing music based on their maps, middle school dance students, studying the original Nijinsky choreography and concepts of biology and movement from their science classes, created original choreography to go with the original music. The details of the students' processes of working with loop-based creative music software to create their original interpretation of Stravinsky's *Rite of Spring* will be presented.

Cranbrook Art & Music Project

The Cranbrook Art & Music Project was an interdisciplinary project in visual art and music where my students studied visual art and musical exemplars (Picasso's *Guernica* and composer Adolphus Hailstork's *American Guernica*) investigating how visual artists and musicians use the tools and concepts of their domain to express and create. After studying these two works, students spent three class periods at a local modern art museum to study works in the permanent collection. With facilitation from the museum educational staff,[2] students selected a work to study and analyze. Guiding questions posed to the students included:

1. What is the medium and which visual concepts/elements are evident in this work?
2. What was the artist's process of creation?

3. What was the artist's expressive intent?
4. What parallels to musical concepts and processes exist?
5. How might you realize aspects of this artwork musically?

After the gallery sessions, students returned to the computer lab and started creating original musical works inspired by the visual artworks studied in the gallery. Students exhibited a wide variety of approaches to creating their compositions, drawing on affective, structural, and conceptual aspects of the artworks. Upon completion, students' compositions were exhibited next to the artworks in the permanent collection at the gallery. Gallery attendees were able to listen to the students' musical compositions via headphones through the use of a portable USB *Flash* MP3 player while viewing the work of art. Attendees also were provided a narrative of the students' creative process and artistic intent (written by the students themselves) as they viewed and listened.

Lessons Learned

Through a naturalistic, qualitative inquiry (Eisner, 1998) of my and my students' classroom experiences, several salient characteristics of successful lessons emerged. These became evident through observations of my students' creative process working with technology during these projects, informal classroom interactions and interviews with students, analysis of students' compositions, and my own reflective experiences as a teacher learning and teaching alongside my students.

Reconsidering the Nature of Initial Composing Experiences with Technology

Evident at the beginning of each semester was the large amount of time students needed to familiarize themselves with the sonic and technical possibilities of the software. Students approached their initial composing experiences using a wide variety of composing strategies. This is supportive of the findings of Stauffer (2003) and Nilsson and Folkestad (2005) that students naturally use a variety of strategies when creating music with technology, especially during initial creating experiences. From this, it seems that an emergent approach to furthering musical understanding tailored to the needs of the students, rather than prescriptive, uniform approaches, may be a more effective way to scaffold student composing throughout initial composing experiences. The writing workshop strategies of Atwell (1998) and Calkins (1994) seem particularly suited to this approach.

Creating a Culture of Collaboration in the Technology Lab

As suggested by social constructivist approaches to music learning (Wiggins, 2001), creating opportunities for students to collaborate throughout the process of composing were found to be valuable. The most successful projects allowed students to work in flexible-size groups or individually, encouraged peer scaffolding and dialogue on and through creative process, and involved

regularly sharing and publishing student work in class and through online media galleries. As well, the technology lab environment, where computers were arranged around the perimeter of the room, seemed to promote student collaboration through allowing free movement around the room. As classes progressed, student collaboration and support seemed to become an integral part of the classroom culture.

Designing Composing Experiences around the Unique Characteristics of the Technology

The projects that seemed to be most engaging to the students were those in which composing experiences were organized around the unique capabilities of the technology and were connected to the music of the students' sound worlds. This engagement resulted in more time spent composing and successively more complex and musical compositions. Often, the structural characteristics of what students naturally created during the projects described in this presentation surpassed the complexity and sophistication of basic song and western classical forms typically used to structure middle school composing experiences with technology. Additionally, the unique visual characteristics of the loop-based software provided clues for assessing musical understanding and served as prompts for discussing and facilitating students' creative process when composing.

Choosing Musical Exemplars for Teaching

I found myself almost exclusively drawing on popular music exemplars to help students refine and further their compositions. Many of the students created music reflective of popular music genres during opportunities for free composition. This seems appropriate given that the technologies students were working with are similar to those used to record and produce today's popular music. Similar to findings by Nilsson and Folkestad (2005), the use of modern artworks and music compositions as models for composing within the curricular environment of our class also resulted in rich, complex, and personally satisfying musical compositions.

References

Atwell, N. (1998). *In the middle: New understandings about writing, reading, and learning.* (2nd ed.). Portsmouth, NH: Heinemann.

Calkins, L. (1994). *The art of teaching writing* (2nd ed.). Portsmouth, NH: Heinemann.

Eisner, E. (1998). *The enlightened eye: Qualitative inquiry and the enhancement of educational practice.* Upper Saddle River, NJ: Prentice Hall.

Nilsson, B., & Folkestad, G. (March, 2005). Children's practice of computer-based composition. *Music Education Research, 7*(1), 21-38.

Savage, J., & Challis, M. (2002). A digital arts curriculum? Practical ways forward. *Music Education Research, 4*(1), 7-24.

Savage, J. (2005, March 31). Information communication technologies as a tool for re-imagining music education in the 21st century. *International Journal of Education and the Arts.* Retrieved March 31, 2005, from http://ijea.asu.edu/v6n2/.

Stauffer, S. (2003). Identity and voice in young composers. In M. Hickey (Ed.), *Why and how to teach music composition* (pp. 91-111). Reston, VA: MENC.
Wiggins, J. (2001). *Teaching for musical understanding*. New York: McGraw-Hill.

Endnotes

[1]I am grateful to my colleague and director of dance at Cranbrook Schools, Gina Buntz, for her expertise and collaboration throughout this project.

[2]I am grateful to Elena Ivanova, educational curator of the Cranbrook Art Museum, for her expert tours and teaching during our museum visits.

Progress and Promise in K-12 Music Education: Is Technology Making a Difference?

Jack A. Taylor (Moderator), Florida State University (retired) and the University of North Carolina–Greensboro
Nancy H. Barry, University of Oklahoma
Sara Hagen, Valley City State University
Jane Kuehne, Auburn University
Scott Lipscomb, Northwestern University
Gregg Norris, Sidney High School, Oneonta NY

[Editor's note. This panel presentation is incorporated into the article in this issue of the JTML titled **"SPECIAL REPORT ON THE STATUS OF TECHNOLOGY IN MUSIC EDUCATION"**].

It's Free! A Critical Review of Free Music Education Resources on the Internet

Nancy H. Barry, University of Oklahoma

One need only type the word "free" into a search engine along with other terms such as "clip art," "sheet music," or "lesson plans" to reveal an overwhelming list of potential resources. Unfortunately, many of these offers of free materials for music educators turn out to be little more than appealing ways to lure potential customers into commercial web sites. Many, many hours can be invested in reviewing these sites before one locates valid educational materials available at no cost and with "no strings attached."

This presentation provides a critical review of free online materials for music educators with suggestions and recommendations for practical classroom applications. Topics include clip art and graphics, MIDI and MP3 files, sheet music, lesson plans and other instructional materials, and other resources for music teachers and students.

Miscellaneous Teaching Materials

Practice Spot (http://www.practicespot.com)

This is a nice source with a good variety of teaching materials for beginning to intermediate level students. Includes articles, and many "printables" (nice quality graphics, Scorch files) such as scale sheets and rhythm exercises. There is a commercial component (books for sale) but this aspect is not overplayed. Most free teaching materials are available without registration.

MERLOT (Multimedia Educational Resource for Learning and Online Teaching) (http://www.merlot.org)

The collection of music materials is not as extensive as in other academic areas (a recent search for "music" produced 164 materials matches), but MERLOT still is an excellent resource for online music teaching materials. I especially like Peer Reviews and Comments, which save time by helping one select the best sites.

K-12 Resources for Music Educators (http://www.isd77.k12.mn.us/resources/staffpages/shirk/k12.music.html)

This site features a very extensive list of links to resources for music educators and students.

MIDI and MP3

MIDI and Music Technology Resources for Teachers (http://www.isd77.k12.mn.us/resources/staffpages/shirk/midi.html)

This site offers an extensive list of links for free MIDI, MP3, and other music technology resources. Lots of sites with free MIDI files are on this list along with other resources. Some links did not work, so I am not sure how often the site is updated, but it is still a great resource.

Theory and Ear Training

MusicTheory.com (http://www.emusictheory.com/)

This is a nice site with a wide variety of drills including rhythm reading, rhythm dictation, note reading, key signatures, chord drills, etc. Registration is required to participate in the Discussion Forum, but most drills are available without registration.

Big Ears (http://www.ossmann.com/bigears)

Self proclaimed as "the Original Online Ear Trainer!," Big Ears features a wide variety of ear training exercises via Java applets. Excellent site, no registration required. Applets may be slow to load on some systems.

Good Ear (http://www.good-ear.com)

This is a noncommercial site that offers a good variety of ear training exercises. Registration is not required. It is a free, noncommercial service, but there is a request for donations.

Sheet Music

The Sheet Music Archive (http://www.sheetmusicarchive.net)

This is a nice collection of free public domain editions of public domain classical music in PDF format. The site restricts downloads to two per day. (The entire collection is available on CD for $19.95.)

Sheet Music Online (http://www.sheetmusic1.com)

It has free PDF files of public domain classical sheet music, some with practice notes available. The files are of good quality; no registration required.

Music for the Nation: American Sheet Music (http://memory.loc.gov/ammem/mussmhtml/)

This Library of Congress site contains more than 62,500 pieces of historical sheet music registered for copyright: More than 15,000 were registered during the years 1820 through 1860 and more than 47,000 were registered during the years 1870 through 1885. This extensive historical collection includes popular songs, operatic arias, piano music, sacred and secular choral music, solo instrumental music, method books and instructional materials, and music for band and orchestra.

Music History and Composer Biographies

The Classical Music Navigator (http://www.wku.edu/~smithch/music/top.htm)

This database consists of five collections of material: (a) a master, alphabetically arranged "Composers" list (including basic biographical data, major works, and influences) of 444 composers, (b) a "Basic Library" list of works from the master list rearranged by musical genre, (c) a "Geographical Roster" in which the names of the 444 composers are listed under the names of the countries that they are associated, (d) an alphabetically arranged "Index of Forms and Styles" listing the names of composers associated with each entry, and (e) a music "Glossary."

The Classical Music Pages (http://w3.rz-berlin.mpg.de/cmp/classmus.html)

This web site provides a nice overview of music history, composer bios (including portraits and short sound clips), descriptions of various musical forms, and a music dictionary.

Pianopedia (http://www.pianopedia.com)

This is an extensive database with 863 composers, 5210 works, 14,866 movements or excerpts representing 50 countries. Scanned score excerpts provide a useful preview for students or teachers. One may browse the database without registering. There are a couple of commercial links, but they are small and do not interfere with browsing the database.

Composer Biographies (http://www.cl.cam.ac.uk/users/mn200/music/composers.html)

This is a nice site with some useful links–but this list of composers is not as extensive as the ones listed above.

Music Notation Software

FINALE Notepad (http://www.finalemusic.com/notepad/index.asp)

Sure, there's a commercial agenda behind this free sample (a scaled down version of FINALE), but it still is a terrific free download. FINALE Notepad is a great little notation program for students or even for teachers wishing to print high quality music notation. New features with the 2005 version include guitar and bass TAB. The download is free or a CD-ROM is available for $19.95.

Graphics

Music Graphics Galore (http://www.intcon.net/~songbird/graphics-1.html)

This extensive collection of free music graphics, including animated gif graphics, lines, buttons, backgrounds, and borders is one of my favorites. Registration is not required and the site is wonderful. It is a great collection of free music graphics.

Composing with GarageBand

Ronald A. Hemmel,
Westminster Choir College of Rider University

In January of last year (2004), Apple released a new program as part of its iLife suite of applications. GarageBand combines loop based composition, MIDI-triggered internal instrument sounds and digital recording of live performers in a very accessible, very inexpensive program. In this session, the basics of the program was explained and explored. Students, schools, and instructors who are not yet ready for the expense or complication of major computer music software programs may find GarageBand to be just what they're looking for. If you have ears and can click a mouse, you are ready to start making music with this program.

The GarageBand program is an excellent introduction to using the computer to make music. The interface is straightforward, providing the user with instrument icons, tracks, graphical volume and pan operators and editing windows.

My session was for beginners. The program was described, demonstrated, and new music was created. The following areas were discussed:

1. Program organization
   - Opening splash screen information

- Loops: What they are, how to use them, how they will work for you
- Tracks: What they show, using pan, mute, solo, volume con trols

2. Creating a song with loops
   - Loop selection paths
   - Saving favorites
   - Adding, editing, and extending loops
   - Panning and balancing
   - Export to iTunes (and beyond)
3. Using MIDI
   - What MIDI is (performance information) and isn't (sound)
   - MIDI representation in GB
   - Using the GB keyboard
   - External MIDI inputs
   - Editing MIDI tracks in GB

4. Real Instruments
   - Recording
   - Effects

5. Online resources
   - GarageBand discussion board
   - Shared file sites

Finally, we considered some of the reactions to GB which should "warm the hearts" of all music professionals. My "secret" reason for encouraging its use is the resulting thirst for more musical understanding.

Consider the following (edited) excerpts from posts at the Apple discussion site (Basically, these posts were listed under the category, "This program has inspired me to want to learn more about music. Can you help me"?)

- Now that garage band got my attention, I am interested in learning how to play the piano.

- Now I'm interested in learning some musical theory so I can learn to write a melody over the top of what I'm doing with GarageBand.

- Can anyone suggest some basic ground rules for creating good com positions?

- Can you recommend books on simple theory and composition? I'm looking for something geared towards people like me that want to be able to build music in GarageBand with flow, balance and rea son.

- I played piano for about six years, but I stopped about five years ago (when I picked up guitar). Now I'd like to relearn the piano/ keyboard for obvious reasons.

- My 9 and 11 year-old boys keep asking, "Dad, can I do GarageBand"?

- Now, however, I'm just interested in learning about MIDI–how it works, etc. Can anyone recommend a book or something for some one new to the area?

- Does Apple or any other company plan to offer piano, guitar or other instrument instruction through GarageBand software or some-related product? Is there an endorsed music instruction program currently available?

- I used to play piano many moons ago and with the advent of Garage Band and the keyboard available I wanted to brush up on my skills.

- I can once again get back into composition without the hang-ups of having to be a virtuoso on several instruments (not that I can claim even one instrument).

Using Music Technology to Teach Music Theory Concepts

Geoffrey Kidde, Manhattanville College

Music students encounter what seems to be a contradiction: Music is *creative* expression, but music also contains many rules (or at least concepts that can come across as rules). I have a vivid memory from when I was 11 years old at a music camp. An elderly lady instructed each of the campers in four part harmony (I didn't know one chord from the other), and she always had the same comment for my exercises: "You have broken all the rules." Composing music is an act of creation; and composers have often discovered new ways of putting musical materials together, ways for which the rules have yet to be written, so that one might say they seem also to be breaking rules. But when we come to teaching young people how to compose and arrange music, we start them off by having them create harmonic progressions which follow guidelines that are based on predetermined scales. There is right and wrong in this process; possibilities for creativity are purposefully limited. We might say that we are trying to instill in them an understanding of what materials are available to them, so that eventually they can develop their own unique way of putting those materials together.

The initial impetus for this project was to use music technology to create free software programs that would assist in this process of teaching young people about musical materials and fundamental concepts. Hopefully, these programs will foster a creative approach to the materials. One of

the first ideas that came to me was to create a program that would allow someone to play around with the idea of a scale and scale degrees.

The program I used is called Max/MSP, which is really a blank slate for creating music technology. The current developer of Max/MSP is Cycling74 in San Francisco, and they term Max/MSP a programming environment. Initially Max/MSP was just called Max (after Max Matthews) and it was developed at IRCAM in France in the 1980s by Miller Puckette and others. Its purpose was to allow composers to use computers and music technology with more flexibility in real time. One thing I love about Max/MSP is that when you open up the program you quite literally see a blank window, save for a row of tools or objects you use to create things. Figure 1 shows a blank Max/MSP window.

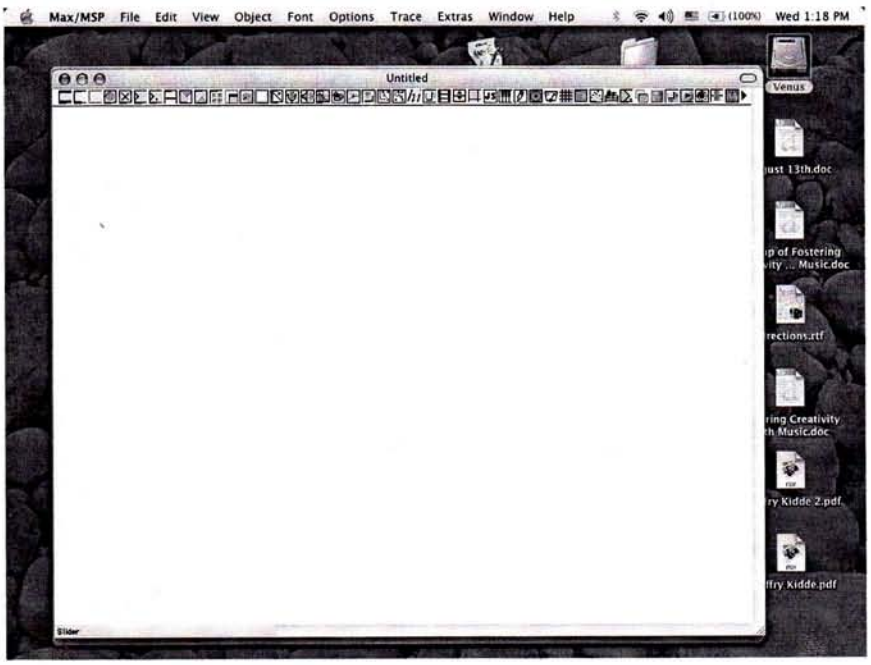


Figure 1. Blank Max/MSP window.

Programs in Max/MSP use objects which get connected on screen. My Scale Maker has a number of objects, but most of the connections are hidden, so as not to confuse the user (see Figure 2).

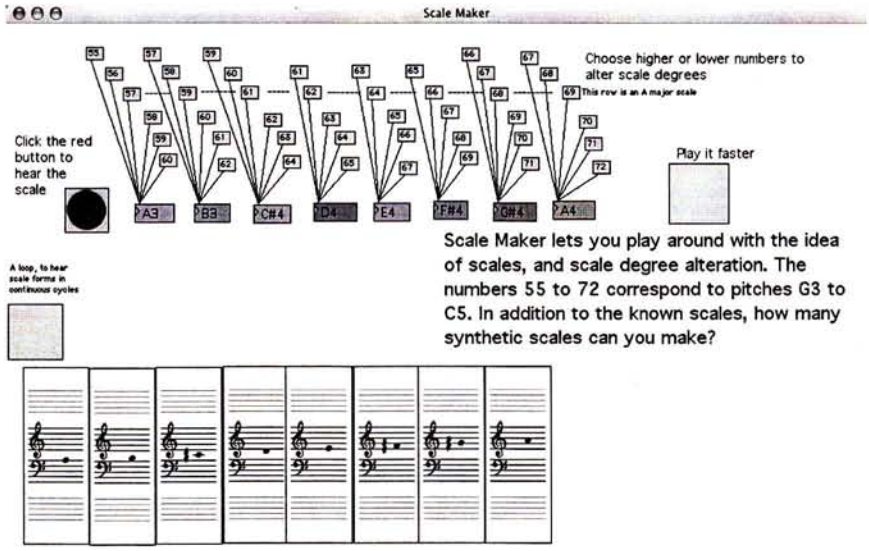


Figure 2. Scale Maker.

As programming in Max/MSP goes, this is actually a simple program, because there is nothing underneath the top level. The third row of numbers will create an A major scale.

Once you have put the scale in place you can hear it by clicking the red button, or you can create a loop to hear it as many times as you like, or you can hear it at a faster tempo. Now I can change some of the scale degrees to create different scales by clicking on different numbers. For instance by lowering the third, sixth and seventh scale degrees I create a natural minor scale. By then lowering the second scale degree I create a Phrygian mode. By returning to the major form, and raising the fourth scale degree I have the Lydian mode. Finally, by lowering the second scale degree, I have a synthetic scale that contains a lowered second and a raised fourth.

Another set of possibilities with this program would be to return to the major scale. While looping it I can restart the scale while the scale is playing the third scale degree, so that I hear the parallel thirds. Expanding on this idea, I could restart the scale with the initial scales on the third and fifth scale degrees, and create the triads associated with the major scale. Eventually I hear alternating A major seventh and B minor seventh chords. By syncopating the clicks and improvising a bit, I can get some other interesting effects. Scales are great fun, and the major scale is still a source of interesting possibilities.

But for that matter we can create the thirds and triads with our synthetic scale, although it is possible to arrange the scale degrees so that some of the results using this method are not triads at all. From the synthetic scale A, A#, C#, D#, E#, F#, G, A, augmented and diminished chords are heard. The two sevenths chords would be an A dominant seventh with raised fifth, and a chord with an A#, D#, F#, and A, which might function perhaps as D# minor chord. Or I could create whole tone or octatonic scales.

Next came the idea of playing with the primary chords and their inversions. Primary chords are often arranged in I-IV-V-I pattern, and that the pattern reflects what we might call the basic function of the three chords: tonic, sub-dominant or dominant preparation, and dominant, and back to tonic for the release of harmonic tension. As we study keyboard harmony, or traditional harmony, we learn how to substitute inversion for the root positions of these chords, and then how to substitute different chords for these primary chords. I wanted to make a program that would create progressions using the primary chords without a preconceived connection to their function. One key to this was the computer's ability to produce random or quasi-random patterns. Figure 3 shows the Progression Maker screen.

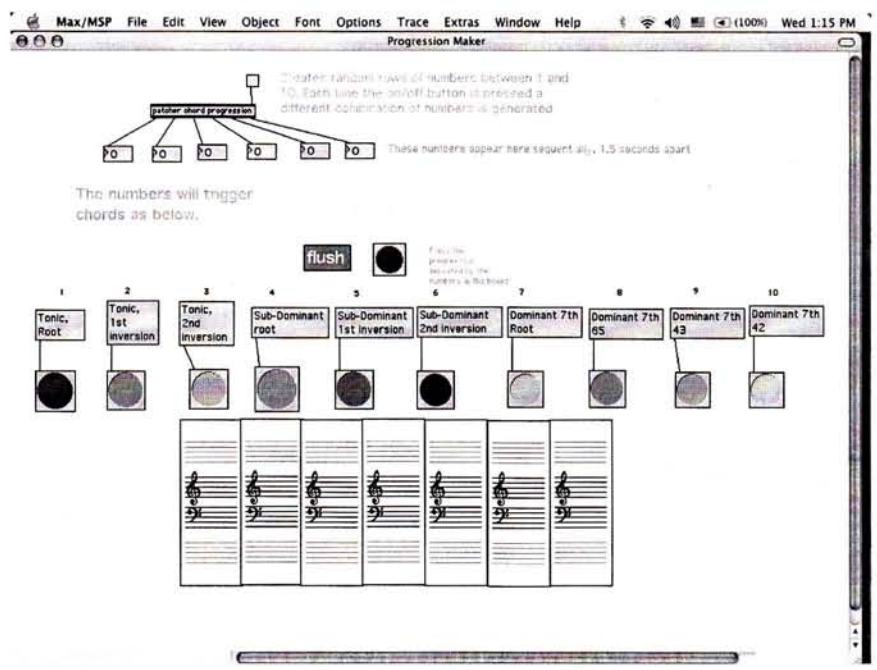


Figure 3. Progression Maker screen.

We can hear the primary chords, each with its root position and its two inversions. The dominant seventh chord appears in the root and the three inversions. What progression maker does, quite simply, is to put the chords in a sequence. The sequence will not repeat any of the chords directly, one after another, although chords are repeated if there are intervening chords. Finally the tonic chord always appears last, whether or not it really feels like a final chord.

Each time you click on the small box at the top of the screen, it will create a different progression. It might eventually repeat a pattern it has already played, but this does not really happen often. One interesting aspect of this is that some progressions work much better than others. This might get someone to thinking about why this is so, and one might begin to discover something about the rules behind primary chord progressions. But if you do find a progression that works well, you might want to repeat it, and play around with it, such as interpolating one of the other chords into the progression. Sometimes this can add coherence. Along with evaluating the progressions that occur, a student of harmony can develop a sense of experimentation with these progressions. The colors of the buttons have no significance other than to give a visual clue as they are played. Finally, I can imagine that this device might be used in a context of a piano class. The notation elements always have the clefs, and there is no rhythmic notation, but perhaps it could be used nonetheless.

In terms of programming, Progression Maker is a bit more complex than Scale Maker, mainly because the computer itself is called upon to create a progression, as opposed to the user creating the scale. The program selects randomly from among the 10 possible chords; there are no rules governing which chords can follow any given chord, except that the tonic chord always comes last, and the same chord does not repeat consecutively. Adding more rules of some sort would be an interesting way to develop the program, but my initial idea centered on the idea of experimentation. Note that we can get to a subprogram within the program by double clicking on the patcher chord progression. This subprogram is what randomly makes the series of chords, somewhat rolling a number on a die–one after the other.

The next step for me was learning to how use the program while creating 20th century and 12-tone music. Twelve-tone music seems to have sort of fallen off the compositional radar, although it certainly was quite significant in the last half of the 20th century. However, I believe the study and creation of 12-tone music can help a student develop a greater sense of the possibilities of harmonic and melodic invention beyond the traditional scales and chords. My Set Maker randomly creates 12 tone rows, and also will automatically generate the 48 other permutations of the initial row (see Figure 4).

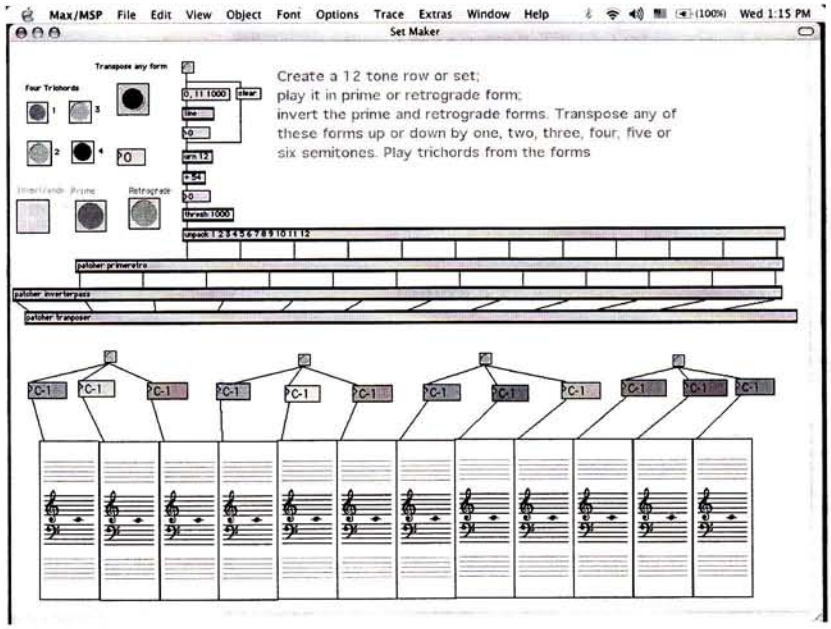


Figure 4. Set Maker.

Set Maker was really intended as a fun way for music students to explore the concept of the row, as used in 12-tone music. It is not something that could write the entire piece, or even really part of a piece. For one thing, the rhythm always is undifferentiated.

Set Maker program will create a series and play it from beginning to end. The first version of the series is the *prime*. It will also play the *prime retrograde*. It can invert either the *prime or retrograde* forms, and it can transpose any of these to any pitch level up or down. In serial composition parlance the 48 possible row configurations are called a matrix. There are twelve transpositions for each of the prime order, retrograde, prime inversion, and retrograde inversion. Each of these can be heard, and the pitch names jotted down for a composition. Maybe the Set Maker can make composing 12-tone music a bit easier. Of course the computer is generating the rows randomly, so that to get an interesting row, you may need to roll the dice a few times. It reminds me that John Cage went to Los Angeles as a young man to study with Arnold Schoenberg. Would this program have something for both of them? And what makes an interesting row? What are the properties of an interesting row? One way to expand the program would be to program ways in which to limit the row, such as to create rows where

certain intervals are favored. This would really be a way to create rules for the Set Maker.

Finally, the program groups three consecutive pitches into four trichords (three note groups, not necessarily triads). With these six elements one can construct simple 12-tone melody and accompaniment figures. One can also hear that the twelve tone row's trichord occasionally falls into traditional sonorities such as triads, or seventh chords chord sonorities (missing the fifth of the chord). Additionally, the trichords sometime produce quartal or quintal harmonies (chords constructed of fourths or fifths respectively). Listening for these chance appearances, and listening to all the possible permutations of the 12 pitches, might give a composition student some fresh ideas about how to create music.

There are many other possible designs which are suggested by these three programs, and I will conclude by saying that these are early versions of the programs. Among the ideas for expanding these programs.

1. For Scale Maker:
   - Increase the possibilities for creating chords out of the scales, so that the user is not relying only on clicking in time
   - Expand the notation element for chords as well.
2. For Progression Maker:
   - Create different voicings for the chords, so that the octave arrangement of the chords changes as well as the order of the chords
   - Include the entire set of diatonically related chords and their inversions
   - Build more rules into the program to increase control of how the progressions are created.

As it stands, the only rules built into the progression are that all the chords are primary, the chords are not repeated immediately, and that the tonic chord is played last. Finally, to let Set Maker allow users to determining more specific aspects of the set, perhaps saying that the set had to create quartal harmonies–or that it would be a fully combinatorial set.

For anyone who would like it, I have copies of these programs which I would be happy to give to you. I plan on posting them (and future versions) on our Manhattanville College Music Technology Webpage.

Using Sound in *Flash* to Enhance the Music Learning Experience: A Demonstration and Workshop

Scott D. Lipscomb, Northwestern University

One of the great challenges facing educators at present is the task of making the classroom more engaging for students. How can we most effectively get students actively involved in the learning process, rather than sitting passively as the instructor lectures on the topic of the day or rehearses that same piece again and again. It is possible that potential solutions to this dilemma might be found in a place considered most unlikely by many educators: video games. I can almost sense the uncomfortable stir when this topic is mentioned, as many begin too regurgitate oft-heard statements such as "Video games are just a waste of time" or "Most games are full of violence." To initiate this discussion, however, I am going to ask that we refrain from such expository statements and focus on what it is, specifically, that young people find attractive about these games, many of which are highly complex in their nested structures, requiring high-level thinking skills to navigate their virtual domains. To put it concisely: "What rewards and satisfaction do video games offer our students that we are failing to provide them in the classroom?"

In order to be successful at playing a game (i.e., to "win"), one must be able to learn and adapt to the rule system created by the designer and apply the necessary problem solving skills to safely navigate through the virtual "world" of the game. This problem solving process often requires impressively advanced critical thinking skills, as various options are considered. This is true whether the goal is to arrange water pipes in the world of *Myst* in order to allow access to a remote location or to determine the best method of hijacking a car in *Grand Theft Auto*. Though we can certainly agree that one goal is more admirable and socially acceptable than the other, similar critical thinking and problem solving skills are involved in each instance.

I would like to propose that we, as instructors, take seriously the charge to make our instructional materials more engaging for our students. Multimedia instructional materials afford one possible means toward meeting this goal. When developing such materials, user interaction can be a designed requirement–unless the user interacts with the interface, nothing happens. Rather than passively watching a slide show, for example, at least a modicum of engagement is forced upon the student. Though we do not yet have definitive proof that technology enhances learning, research suggests that there are few cases in which the presence of technology, when effectively incorporated, has proved detrimental to learning. Likewise, this same research continues to reveal that student attitudes toward the use of technology are consistently positive (Webster, 2002). As every teacher knows, a positive attitude provides a large step in the right direction.

There are, of course, varying levels to which interactivity can play a role in the learning experience. It is possible to design a set of materials that requires the user to click on a button (or provide some other input) before proceeding from one section of the materials to another. This low level of interactivity leaves the user sitting passively between these sequential interactions. Another possibility, in which a higher level of activity is required, would be to require the user to click to initiate an action, click again to stop the action or initiate another action. The much higher level of interaction evident in video games, however, allows the user to interact in a manner that has consequences for future actions, resulting in a series of decisions that ultimately leads either to success or failure. Fortunately, in the context of video games, failure means simply that the user must start the game over or return to a recently saved location within the game, which is quite different from the result of failure in a real life context.

During the past decade, I have integrated technology and student interactivity more and more into my music classroom. I began by simply providing streaming audio and video, allowing students to access sound files 24 hours a day, seven days a week. This kind of start-stop interaction provided a high level of convenience, but lacks the kind of engagement considered necessary for deep learning. For demonstration purposes, using *Director* and *Flash*, I then created numerous animations to illustrate complex concepts like musical form and the Gestalt principles as they relate both to visual and auditory cognition. Most recently, I have begun creating multimedia materials that allow my students the opportunity to navigate from one point to another within a piece of music and to explore the composition in a manner that would not otherwise be possible, rather than enforcing upon the student a linear (i.e., beginning-to-end) listening experience. Some of these same tools can be provided to advanced students to enable them to use their higher level skill set to create their own projects and demonstrations.

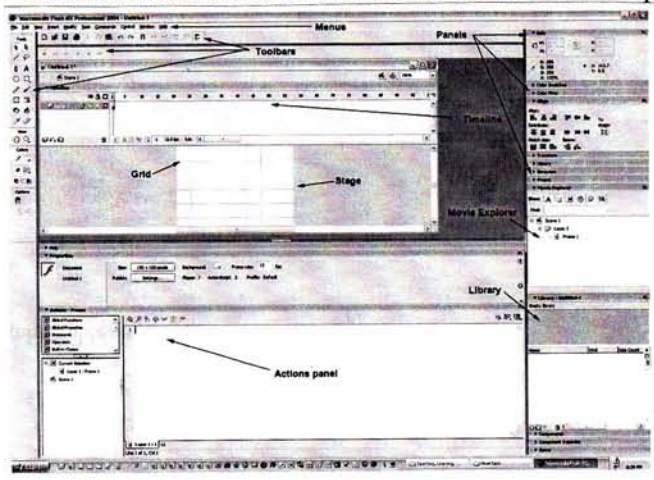
A Tool of Choice: Macromedia's *Flash*

Macromedia *Flash MX 2004* provides an authoring environment for the creation of truly interactive multimedia materials.[1] The free browser plug-in and player available from the Macromedia web site have established a nearly ubiquitous presence on the Internet. Over 97% of computers used to surf the WWW have the plug-in installed, making this program one of the most reliable ways to ensure that users will be able to utilize interactive instructional materials to their fullest extent. *Flash* provides a marvelous balance between user friendliness and complexity. It is possible to learn the program's most basic capabilities and put these to work almost immediately in the creation of interactive instructional materials. For those who are willing to spend the extra time to learn ActionScript, the scripting language that is part of *Flash*, the program's potential is almost limitless. One disappointing limitation is that *Flash* does not include MIDI capability, though it masterfully handles numerous types of digital audio files and substantially re-

duces file size of the web-ready movies using effective compression algorithms.

Flash offers many other attractive and useful capabilities. It allows users to import graphic images, video, sound files, and many other types of media created in other programs or downloaded from the Internet. Many types of objects (e.g., text, ovals, rectangles, lines, customized gradients, etc.) can be created directly in *Flash*, using the Tool palette. These basic shapes and objects can be combined to create professional looking graphics for use in your own animations. In addition, *Flash* comes with several "Common Libraries" to provide many essential tools. The "Buttons" library, for example, contains an impressive array of buttons, sliders, knobs, faders, and other useful objects that can be readily included in your movies. The "Learning Interactions" library contains templates–including complex interactive capabilities–for creating various types of quiz forms: true-false, multiple choice, fill in the blank, drag-and-drop, and others. In addition, the program comes with an impressive set of components with built-in functionality. Some of the most commonly used Components–Macromedia's built-in, interactive objects–include the checkbox, radio button, progress bar, listbox, combo box, text input, and many others.

Like many Macromedia programs, *Flash* uses a "Timeline" metaphor to facilitate the creation of animations. This allows complex animations to be created without necessitating the frame-by-frame creation of each individual image. Instead, the user marks important locations (called "keyframes") along the Timeline and sets the desired location of each object in the movie at this point in time. *Flash* then automatically calculates the appropriate location for every object for every intervening frame, a process known as "tweening." Because *Flash* utilizes vector graphics instead of bitmaps (or raster graphics), the resulting files are much smaller and take less time for online visitors to download. The context sensitive Properties window allows users direct access to the most common attributes of any object selected in the Work Area. Other panels (accessible from the "Windows" menu) allow the user to set alignment, transform and skew objects, mix colors, use pre-made Components, view all objects in the present movie, along with many other possibilities. For those who wish to move to more advanced levels of interactivity, *Flash* provides a powerful programming



language called *ActionScript*, providing many of the same capabilities that previously were only the purview of much more complex and difficult to learn programs like Java and C++.

There are two primary file types with which you must be familiar when working with *Flash*. When creating a *Flash* file (called a "movie") using the *Flash* interface, you work with a ".fla" file–the native format for *Flash* movies. When you "publish" a web-ready version of the movie, however, it is saved in ".swf" format (pronounced "swiff"). It is the .swf file that must be uploaded to your web site, *not* the .fla file. An internet browser (Internet Explorer, Netscape, Mozilla, Safari, Foxfire, etc.) with the *Flash* plug-in installed[2] will readily play back .swf file. In all likelihood, the browser will not know what to do if you link to a .fla file.

Work Flow in *Flash*

As you work with *Flash*, you will find that there are typically a series of steps involved with creating interactive multimedia content. The most common sequence of procedures is outlined below. You can:

- draw or import graphics and sound files into *Flash;*
- transform buttons, independent animations, and any other elements you intend to reuse into "symbols;"
- place movie elements (vector graphics, bitmaps, symbols, etc.) on stage;
- attach actions to buttons, movie clips, or frames on the timeline to make them interactive (in the steps outlined below, you will find that you are strongly encouraged to create two specialized layers in your *Flash* movies: an "action" layer (to hold ActionScript elements) and a "sound" layer (providing a single location for your audio components);
- select a frame, symbol, stroke, fill, or text block on the stage to adjust its properties via different panels.

Below are step by step instructions for each topic covered in the workshop part of this presentation.

Animation Basics

Creating Keyframes:

- Insert→Timeline→Blank Keyframe: to create brand new content in a new frame.
- Insert→Timeline→Keyframe: duplicates contents of previous keyframe.

To add a Motion Tween animation, follow these steps:

- In the layer containing the object(s) you wish to animate, simply click on the frame where you want your animation to begin.
- When you create a new movie, the first frame of Layer 1 automatically has a keyframe; otherwise, you can add one

(Insert→Timeline→Keyframe). If your objects are not yet on the stage, *ensure that a keyframe on the appropriate layer of your movie is selected*, then drag the object(s) from the Library onto the Stage at the location you would like for them to appear.

- Click on the final frame where you would like the animation to end, and then select Insert→Timeline→Frame.
- Right-click (Win) or Control-click (Mac) the first keyframe of this animated sequence to open the pop-up menu and select "Create Motion Tween." At this point a dashed line will connect the first frame of your animation to the last in the Timeline. This lets you know that the process is not yet complete.
- Click once again on the final frame of your animation and move the object(s) to the location where you would like it (them) to be at the conclusion of this animated sequence. Now the dashed line should have changed to an arrow pointing from the first frame of your Motion Tween animation to the last frame.

Sound in *Flash*

Importing Sound and Using It in Your Movie

- File→Open, then browse to & select the "01soundCrash_template.fla" file.[3]
- In the sound layer of the Timeline, click on the placeholder frame where you want your sound file to play (in this case, frame 20), then select Insert→Timeline→Keyframe (or F6). You must *always* add a keyframe to the frames in your timeline where you want a specific event–like initiating playback of a sound file–to occur.
- With the frame containing the newly added keyframe selected on the sound layer of the Timeline, *drag the sound file* (CRASH.WAV) *from the Library panel onto the Stage.* Notice that a representation of the sound wave appears in the frame to confirm that a sound has been added.
- While the frame with the sound file is selected, notice the Proper ties window now contains additional options for manipulating the sound.
    - o From the Effect drop-down box, select Custom and notice how you can control the playback of your sound file; to get to this same window, you can simply click on the "Edit" button to the right of the Effect drop-down.
    - o Set the "Sync" mode to "Event" and repeat "0" times. Learn about the different Sync types (Event, Start, Stop, and Stream) to ensure that you get the most out of using sound in your *Flash* movies.
- Control→Test Movie

Important capabilities to know

- Import a sound file to your library: File→Import→Import to Library, and then browse on your hard drive to select a sound file.
- View the content of your current movie's Library:
  - Window→Library (or CTL+L)
- Notice that after importing a sound file, it is automatically added to your movie's Library.
- If you click on the "Information" icon at the bottom of the Library window, you can determine the appropriate level and type of compression for your sound file.

Adding Sound to a Button

- File→Open, then browse to, and select the 02soundButtons_template.fla file.
- Double-click the "Play" button to enter "Symbol Edit" mode; alternatively, you can select Edit→Edit Symbols.
  - Just above the very left hand side of the Timeline, you should now see the name of the button upon which you are operating to the right of the Scene 1 label.
  - Note that the Timeline for button symbols is different than the movie timeline, containing only four frames: one representing each button state. When you create your own buttons, in addition to the obvious button states, you must also make sure to designate some area of the button as the "hit" area (i.e., the area that responds to user interaction).
- Decide which states (up, down, and/or over) to which you wish to attach sounds; for the current example do the following:
  - Add a layer (Insert→Timeline→Layer) and label it "sound."
  - If your movie's Library window (*different from* the Common Library window) is not open, then Window→Library (or CTL+L).
  - In the *sound* layer, click on the keyframe representing the state to which you wish to add a sound (in this case, the "Down" state), and then drag the appropriate sound from the Library to the Stage. When creating your own buttons, you would repeat this step for every state within which you wish to add a sound.
  - To exit Symbol Edit mode, click on Scene 1 above the up per left corner of the Timeline panel.
- To create interactivity that will allow a user to halt sound playback:
  - Single-click on the Stop button.
- To create interactivity that will allow a user to halt sound playback:
  - Single-click on the Stop button.
  - While the button is selected, add the following code to the Actions window:

```
on (release) {
    stopAllSounds();
```

}

- Control→Test Movie

Important capabilities to know

- Add a button from the Common Library to your movie:
  - Open the Buttons Common Library panel (Window→Other Panels→Common Libraries→Buttons).
  - Select the button you want to use from the Common Library and drag an instance onto the Stage; the button is automatically added to your movie's Library.

Control a Sound Using Frame Navigation

This method takes advantage of the Start and Stop sound Sync modes, initiating and terminating playback simply by navigating to an appropriate frame in the movie using the gotoAndStop() command when one of the buttons is pressed.

- File→Open, then browse to & select the "03playSound_frame Navigation_template.fla" file. Note that a sound file ("Echo Flute 9.5.wav") has been imported into the movie's Library and the following layers exist on the movie's Timeline:
  - Labels–used below to navigate using ActionScript commands. Two labels have been provided for you:
    - "stop" in frame 1
    - "play" in frame 15[4]
  - Actions–where frame scripts can be typed.
  - Main–the name given to the layer containing the buttons.
  - Sound–where the sound file will be placed.
- Click in frame 1 of the actions layer and notice that "stop();" has been typed into the Actions window (Window→Development Panels→Actions). The purpose of this script is to avoid automatic playback of your movie immediately when it is loaded, turning complete control over to user interactions, as determined by ActionScript.
- In the sound layer, add a keyframe (Insert→Timeline→Keyframe) at frame 15.
- Click on the keyframe in frame 1 of the sound layer and drag the sound file from your movie's Library to the stage, then click on the same keyframe a second time and select Stop from the Sync drop-down box in the Properties window.
  - Entering a frame containing a keyframe set to the Stop Sync mode causes the sound file, if playing, to halt playback.
- Now click on the keyframe in frame 15 of the sound layer and drag the same sound file you dragged to frame 1 from your movie's Library to the stage, and then click on the keyframe in frame 15 a second time and select Start from the Sync drop-down box in the Properties window. Entering a frame containing a keyframe set to

the Start Sync mode causes the sound file, if not already playing, to initiate playback.

- The final step is to attach a basic ActionScript to the "release" event for each of the two buttons, causing the playback head of your movie to relocate to the appropriate frame in your movie, as determined by the frame label identified in the gotoAndStop() command.
  - Click the Play button and type the following text into the Actions window:
    on (release)
    {gotoAndStop("play");}
  - click the "Stop" button and type the following text into the "Actions" window:
    on (release)
    {gotoAndStop("stop");}
- Control→Test Movie

Using *Flash's* Sound Object to Instantiate and Control Playback

This method for integrating sound into your *Flash* movies utilizes the capabilities of the **Sound** object provided with *Flash*. In this example, we are going to control sound playback completely through the use of ActionScript, so we need to take a few steps at the outset to inform *Flash* about our sound file and how we will refer to it. This is accomplished by creating a "Linkage":

- File→Open, then browse to & select the 04soundObject_template.fla file.
- Right-click (Windows) or Control-click (Mac) on the name of the sound file ("eurotechno.mp3") in your Library[5] and select "Linkage . . . " from the pop-up menu that appears.
- In the Linkage Properties dialogue box,
  - Type "eurotech" into the Identifier textbox.
    - This is the name we will use in ActionScript to refer to this specificsound file, though, in this instance, the Linkage identifier is very similar to the MP3 filename, this is not required.
  - Place a check in the "Export for ActionScript" box.
  - Place a check in the "Export in first frame" box.
- We now have taken all the preliminary steps necessary to make the sound file accessible to our ActionScript code, so we can begin the process of instantiating a Sound object and attaching a sound file to it, after we accomplish some basic set-up tasks.
- Setting up our movie:
  - Notice that the movie is 10 frames long and consists of three layers (labels, actions, and buttons); also note that there are frame labelson the "labels" layer at frames 1 and 10.
    - Frame 1: "initialize": The purpose of this frame is simply to carry out some basic initialization steps that only need to

be carried out one time at the very beginning of the movie, then cause the play back head to navigate to a later frame where user interaction will occur.
        - Frame 10: "playback"
- Add the initialization code:
    - Click on the keyframe in frame 1 of the actions layer.
    - Type the following text into the Actions window:
        - var currLocation = 0;
        - var mySound = new Sound();
        - mySound.attachSound("eurotech");
        - gotoAndStop("playback");
    - Here is the purpose of this code:
        - First, we create a variable called currLocation that is used to track the current position (in milliseconds) within the sound file.[6]
        - Second, we create a Sound object called mySound.
        - Then we use the attachSound() method to attach the sound file in the movie's Library–remember that we set the Identifier field to eurotech when we created the "Linkage" above–so that any command sent to the mySound object acts on the eurotech sound file.
        - Finally, we use the gotoAndStop() command to send the movie'splayback head to the playback frame label, where all control will be turned over to the user via ActionScript attached to the buttons on the Stage.
- Add ActionScript to the buttons on the stage to control sound file playback.
    - Select the Play button and type the following code into the Actions window:[7]
      on (release) {mySound.start(currLocation / 1000);}
    - Select the Pause button and type the following code into the Actions window:
      on (release) {currLocation = mySound.position; mySound.stop();}
    - Select the Stop button and type the following code into the Actions window:
      on (release) {mySound.stop(); currLocation = 0;}
- Control→Test Movie

Using Streaming Sound

When you want to ensure that audio-visual synchronization is maintained within a *Flash* movie, your best bet is to use the Stream Sync mode, as described below. To see how this works in a very basic way, follow the steps below. After the general concept is understood, you look at–and get more meaning from–the set of musical form templates available on the author's web site.

- File→Open, then browse to & select the 05_I-vi-IV-V_template.fla file.
- Using the steps followed previously, import a digital sound file of a composition that incorporates the I-vi-IV-V progression into your movie's Library (examples: "Mr. Postman," "Stay," and "Runaround Sue").
- Select frame 1 on the "music" layer, then drag the sound file from your movie's Library onto the Stage.
- Click on frame 1 of the music layer a second time, and then select Stream from the Sync drop-down box in the Properties window.
  - Notice that the representation of the digital audio file contin ues from frame 1 until the end of the sound file (which can be *thousands* of frames, depending on the frame rate of your movie and the duration of your sound file).
  - Also notice that when you click and drag the playback head (the red rectangle about the Timeline), you can hear the con tent of the sound file as it passes by.
- By creating other layers in your movie to contain objects, animations, interactive elements, you can synchronize, as precisely as you like, the appearance and/or movement of objects on the stage with your audio sound file.

There is a considerable potential inherent in this method for use in the music classroom, as demonstrated by the form templates available from my web site. Simply point your web browser to: http://www.lecafeamericain.net/faculty.htm, then click on the "stuff" button to navigate to the appropriate page containing a list of available downloads. At the time of NSMIT 2005, *Flash* templates were available for the 12-bar blues, AABA, and I-vi-IV-V progression.[8] On that same page, you will also find BubbleMachine™, a tool–free to all educators–for the creation of bubble charts to allow point-and-click navigation and interactive exploration of any musical composition that has been saved in MP3 format. The form templates require that the user has *Flash MX 2004* (or later), but the BubbleMachine™ program requires only the free *Flash* Player. Documentation and basic tutorials also are available from the web site. The author hopes that you will find these templates and tools useful in creating an interactive environment for your students to explore the world of musical sound.

References

Lipscomb, S.D. & Walls, K.C. (in press). Multimedia authoring. In S. Watson (Ed.), *TI:ME Technology Guide*. Hal Leonard Publications.

Webster, P.R. (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*, pp. 416-439. NY: Oxford University Press.

Endnotes

[1]You can download the free Flash plug-in from: http://www.macromedia.com/shockwave/download/download.cgi?P1_Prod_Version=ShockwaveFlash.

[2]The opening paragraphs of the workshop section of this paper are excerpted, in abridged form, from a forthcoming chapter (Lipscomb & Walls, in press) to be published in the *TI:ME Technology Guide*.

[3]All templates used for this presentation are available from the author's web site. Simply point your Internet browser to: http://www.lecafeamericain.net/faculty.htm. From Scott Lipscomb's home page, click on the "stuff" button, and then click on the link to materials related to NSMIT 2005.

[4]To add your own labels, simply select a keyframe in the Timeline and type the label text into the Frame Label textbox in the Properties window.

[5]Remember, if you do not see the Library window, you can always make it visible by selecting Window®Library (or Control+L)

[6]Using this value allows us to cause the buttons to behave exactly as the user expects from her or his experience with CD and DVD players. Notice, as you study the ActionScript for each of these buttons that the Start button calls the Sound object's start() method, using the value of currLocation (divided by 1000; see next footnote for an explanation) as the starting location. When a user presses the Pause button, the current position within the sound file is loaded into this variable before playback is halted, so that when the Start button is pressed, playback is initiated from the point where the sound file left off. If, however, the Stop button is pressed, the value of currLocation is set to "0", so that when the Play button is pressed, playback starts from the very beginning of the sound file.

[7]There is an inconsistency in the manner in which time is referenced from within the Sound object. All the objects methods use milliseconds as the unit of time *except* the start() method, which, for some strange reason, uses *seconds* instead. Beware of this discrepancy when you are creating your own movies that incorporate the built-in Sound object. This is the reason that the currLocation value must be divided by 1000.

[8]There also is a template for the Sonata Form that was created in *Director* available for download from the same web page. More templates will be added, as time allows.

Sound Ideas for the Internet

Christopher Brown,
Avoca School District 37, Wilmette, Illinois

This presentation included a step-by-step explanation and demonstration of how to add sound to your web page from a variety of different formats and programs (QuickTime, Real Audio, Garage Band, Finale etc). Manipulating recordings of school groups for easy and multi platform web distribution primarily using Dreamweaver was part of the focus. Student compositions through notation software was exported for web playback. Additionally, we discussed the pros and cons of each sound format for audio quality and also for their ease of use on the Iinternet. Finally, a packet was distributed that includes common term glossary and links for further information on related topics.

The MUSE Project: An Online, Customized Entrance Placement Exam for Music Theory

Susan Piagentini, Northwestern University

The use of assessment tests in music theory is standard in numerous universities and colleges across the country. Many schools currently rely on the AP music theory exam. This test includes aural, visual, and written components; however, it may take months for the student and the school to receive scores from the College Board Testing Service. Because of this problem, many universities and colleges are creating their own placement tests to evaluate entering music students' skills. Many of these tests are still on paper and may or may not include an aural component. There have been several studies conducted which focus on the design of computer based placement tests in music theory (Coleman, 1990; Smith, 1994; Murphy, 1999). The most recent study by Murphy indicates that "the exam predicts students' grades in theory class and makes the exam much more useful for both the student and the teacher." Although the above tests have proven successful, there has been no research conducted on a web based music theory placement test.

This fall we began a study funded by a 2004 SBC Faculty Fellows Grant. Over 40 universities and colleges were contacted regarding their use of entrance exams. The data collected included information from large universities with substantial music departments such as Indiana University, Peabody University, University of North Texas, and Florida State University to smaller, private liberal arts colleges such as Meredith College, Franklin and Marshall College, Centenary College, and William Patterson University. The responses from these colleges and universities indicate that the majority of all programs utilize some type of placement exam for their incoming majors. However, the skills expected of each incoming student vary greatly. Some programs expect minimal skills; others are testing more advanced skills based on their entering students' past performances and experiences.

Based on our findings, it is apparent that there is an increased need for a tool to better assess students' placement in undergraduate theory. Because of the variance in students' previous musical experience, instructors must currently spend extra time with incoming students evaluating their knowledge in the form of paper tests and individual meetings. The development of an online assessment test in music theory not only will allow instructors to more closely evaluate student knowledge of music theory, but also will allow students to have prompt immediate feedback on their deficiencies in music theory. An online assessment test also will give an immediate recommendation for the course in which students should enroll. Students would be enrolled in the appropriate section, a section that includes material that will challenge and satisfy the student. Through this, students

will be more fulfilled with their learning experience and student retention in music theory courses will be greatly improved.

It is imperative that the placement exam be pedagogically sound as well as usable, functional, and aesthetically pleasing. There are many different principles that need to be addressed when designing web evaluation or instruction, such as aesthetic appeal, authority and authenticity, completeness, consistency and layout, error support and feedback, and accessibility. In order to ensure effective online application development and focus on the above-mentioned principles, the Center for Teaching and Learning (CTL) was consulted regarding examples of sound online instruction.

To answer the needs outlined above, the basic organization of the online, customized placement test is described below.

1. Structure: The exam platform is a menu based, customized site. It allows instructors to personalize the look of the page and the content of the exam. Professors can control the following variables:
   - *Individualized color graphic schemes by selecting school color for the template and adding a .gif of their school logo* (Instructor also can display their school names and contact information.)
   - *Number of questions they would like to include in various concept groups for both written and aural theory* (For example, note reading, key signatures, intervals, triads, seventh chords, melodic dictation, harmonic dictation, error detection, etc.)
   - *Levels of concept group examples* (For example, Scales, Level 1: would include only major scales; Level 2: major and minor; Level 3: modes, octatonic, whole tone, etc.)
   - *Timings for each question category*
   - *Number of hearings for aural examples*
   - *Reporting of students' results immediately sent to instructor via e-mail*
2. Students can see results upon completion, see percentages of strengths and weaknesses, and receive registration information as determined by the instructor.
3. Score format: A total percentage for all components, or separate score categories.
4. Technical Features: Randomly selected questions from a database of each category. Questions are multiple-choice, using radio buttons for answer input. The musical examples used are .gif files; the audio uses an embedded RealAudio play controller.

The test will be available in multiple formats: as an online authoring tool for instructors; as a review application for students; and as printed format for those who prefer written exams.

Currently we are in the second stage of development of the test. We are sending sample test questions to the study participants, asking for feedback

on the type of questions and the personalized template setup. From these results, we will continue to develop the growing database of questions for each category. To date, we have 135 questions in each of five categories. There will be more categories and questions by the beta release. Recently we released the preliminary test beta version to a selected group of schools for testing in the fall of 2004.

The goal for this project is to place incoming students in a course that will challenge them in the field of music theory. By enrolling in the appropriate course, students will be able to go further in their study of music theory. It is expected that the students will readily use the tutorials and placement exam at the beginning of the semester and will appreciate the immediate feedback and recommendation given. Students will also be given an overview of their musical deficiencies at the completion of the test and will be able to focus on these weaknesses throughout the upcoming semester.

The online test format will free instructors from the numerous hours spent grading written tests and meeting with individual students during new student week. One of the personalized options would provide placement information directly to students so they can enroll for the appropriate section sooner. After using this placement test, a school may use their individual data to evaluate of what types of courses may be needed in the future to accommodate for the skill level of incoming students. By using such an online tool, instructors will also be made aware of incoming students' learning styles. The incorporation of a theory placement test would also affect curriculum design for all of the theory courses. It would allow instructors to improve on their pedagogical approach, thus creating a sound learning environment for each student.

The preliminary results of the study make it clear that the individualized format of the test will be invaluable to instructors from a wide range of programs. It provides an online avenue for instructors to create and administer the test. The "do-it-yourself" web platform with ready-made, cutting edge technology will streamline the process of developing an exam. It will save the instructor numerous hours in planning time, as the musical examples and formatting are complete. They can create a personalized site for their program with the click of a mouse regardless of their own technical skills.

The future of the site will expand to enable instructors to input their own musical examples as .gif files, and MIDI files for listening examples. It will include simple tutorials for authoring new questions, and the end result would be an ever-growing database of questions for all instructors who use the site. With some monitoring and site membership, this could prove to be an invaluable resource for instructors across the country to share ideas, compare their program to other similar schools, and raise the level of skill expectations in their own program.

The Sibelius-Reason Alliance Topic: Beyond Notation and Synthesis

Ernie Jackson,
Queensboro Community College, Wagner College

When using Sibelius 3, people often are discouraged by the quality of the sound card. While Sibelius 3 does come with Kontakt Silver (upgradeable to Kontakt Gold), the method can become taxing on the system, and may not completely offer more electronic sounds.

Sibelius 3 can be used with *any* software synthesizer that is less taxing and more robust in its approach to playback. Using a very simple piece of "free" software, Sibelius 3 can be used with a powerful program such as Propellerheads Reason 2.5. The advantage here is the use of high quality sampled sounds (many available for free online), effects, and synthesis training on a very sophisticated level. I have found this way of teaching very effective in demonstrating writing and notating phrases for waveforms and other electronically generated tones.

The other advantage to this method is the detailed score layout now available to performers as each section of a piece can be notated with controller values and MIDI information as well as patch notes for Reason 2.5. This is good time to present this information because Reason 2.5 has now been upgraded to support performing in a "live" situation, therefore opening the doors to a great way of performing "synthesis with notation." Importing MIDI files for realistic playback and variations on a "tone" will further enhance the ears of students who have vested interest in sound design.

In the notation, area, I will further explore the teaching of instrument sounds with the Sibelius Educational Suite and how it too can be intertwined with Reason 2.5. For example, Sibelius Instruments can further teach how orchestral instruments should sound, especially when teaching electronic music sequencing. Sibelius Notes can be modified to create worksheets and exams for synthesizer patch selection and composition notes.

To Burn Or Not To Burn: It is More Than An Ethical Question

James Frankel, Teachers College, Columbia University

This session examined the ethical and legal consequences of file sharing over the Internet, specifically students downloading and burning music from Peer-To-Peer File Sharing Services such as KaZaa, Morpheus, LimeWire, etc. A brief history of P2P File Sharing was discussed as well as many of the legal cases that surround this very controversial issue. Copyright law was

discussed as well as what music educators can and cannot do with regard to copyright law and burning.

Composing and Transcribing Using Garage Band

Paul S. Carter, SUNY College at Oneonta

Currently I am developing a new pedagogical approach in our curriculum at SUNY Oneonta as a part of our Music Theory coursework in the Music Industry program. In this program, we teach many students who have had little formal musical training. My hope is that this pedagogical approach will apply not only to other college teachers who have students of this level, but also to grade school teachers whose students likely would be less experienced in the theory knowledge and perhaps musicianship needed to complete projects as such. Garage Band, when used in a MIDI input environment, offers a window on the composition or transcription process not achieved in other environments. The latest version of this software, Garage Band 2, includes a scrolling notation of the work completed in the "mix" window. Although it does not print this notation, the visual realization increases the educational value of the endeavor. In this presentation, I explained and demonstrated the use of Garage Band and MIDI instruments necessary for the assignment of composition and transcription projects.

The musical design of the projects is very important. The technology of such an approach involves the development of theory and aural skills. For some time, I have been researching and developing transcription as a teaching tool.[1] The marriage of using transcription with MIDI-enhanced Garage Band presents an effective possibility. For transcribing, I will demonstrate how to design assignments that isolate portions of pieces and elements of the texture of various recorded material. Using audio playback, my demonstration will include not only suggestions on the design of the assignment, but also how to address the coaching of completion of it using audio playback and musical referencing with Garage Band. The program as a sequencer acts as an audio "scratchpad" where students can emulate melodic, rhythmic, harmonic and sonic elements of the music; and in recreating these components, hear what comprises it, as well as seeing it on the screen, both in "mix map" format, and in the scrolling notation now included in Garage Band 2. I have always believed that transcription is an extremely valuable musical learning device. Garage Band, when used in this way, makes this process more accessible and more effective.

Going one step further, Garage Band can be used for composing or songwriting. As an assignment, the value of learning theory through doing this comes from evaluating elements observed in other music (perhaps through transcription), designing a compositional plan to utilize or modify these theoretical elements, and executing in the software. An immediate audio-

visual playback affords the student the opportunity to evaluate the process as it unfolds, step by step, if it is desired.

Endnote

[1] In the Fall of 2002, I gave a paper/presentation at the CMS (ATMI) meeting in Kansas City titled, "A Hitchhiker's Guide to the Pitch Space: Teaching Transcription in the Theory Curriculum."