Music Performamatics: Interdisciplinary Interaction

Jesse M. Heines
Dept. of Computer Science
Univ. of Massachusetts Lowell
Lowell, MA 01854
1-978-934-3634
Jesse_Heines@uml.edu

Gena R. Greher
Dept. of Music
Univ. of Massachusetts Lowell
Lowell, MA 01854
1-978-934-3893
Gena_Greher@uml.edu

Sarah Kuhn
Dept. of Regional Econ. & Social Dev.
Univ. of Massachusetts Lowell
Lowell, MA 01854
1-978-934-2903
Sarah_Kuhn@uml.edu

ABSTRACT

This paper describes how a graphical user interface (GUI) programming course offered by the Dept. of Computer Science (CS) was paired with a general teaching methods course offered by the Dept. of Music in an attempt to revitalize undergraduate CS education and to enrich the experiences of both sets of students. The paper provides details on the joint project done in these classes and the evaluation that assessed its effect on the curriculum, students, and professors.

Categories and Subject Descriptors

K.3.2 **Computers and Education** [Computer and Information Science Education]—*computer science education, curriculum*.

General Terms

Management, Design

Keywords

Performamatics, Computer Science Education, Interdisciplinary Programs.

1. THE PERFORMAMATICS CONCEPT

Performamatics is a program designed to address declining enrollments in computer science (CS) [1, 2, 4, 11, 13]. It is an interdisciplinary partnership between the CS, Art, Music, and English departments in the area of exhibition and performance technologies that introduces CS students to real-world applications in areas that interest them as early as possible in the curriculum.

While Performamatics was originally conceived to benefit CS majors by increasing engagement and helping them see the connections between theory and practice, it also exposes non-CS majors to computing at a level that they don't typically see in computer literacy courses. In addition, both sets of students learn to work with others who may not only have a significantly different perspective on the work, but who may also speak about the work in an entirely different "language." Thus, both sets of students benefit from interdisciplinary exposure that reflects the way real projects are developed in the workplace.

SIGCSE'09, March 3-7, 2009, Chattanooga, Tennessee, USA.

To date the Performamatics team has explored two models for interdisciplinary courses. The first is a "synchronized" model for upper-level courses, in which a joint project is done by students in two pre-existing classes [9]. The second is a self-contained course that typically has no prerequisites, following the "Artbotics" model pioneered by Yanco *et al*. [10, 15].

The courses described in this paper falls in the former category and involve an interdisciplinary capstone project. This approach is not new [5, 6, 12], but each campus faces its own challenges in implementing such projects. By describing our experiences, we hope to help others think about how they might structure interdisciplinary experiences on their campuses.

2. THE SYNCHRONIZED COURSES

2.1 The GUI Programming Course

The CS course involved in this collaboration was the second semester of a two-semester senior capstone project sequence. This course focuses on the object-oriented (OO) nature of graphical user interface (GUI) programming. Exercises are done in Java, but other OO implementation languages and systems (such as Flash) are also discussed. The idea is to teach the underlying concepts in GUI programming so that students learn what they need to know to do quality GUI programming in any OO language or using any OO GUI application programmer interface (API) [8]. Course websites for GUI Programming I and II containing the syllabus, lecture notes, assignments, and references for these courses are linked from teaching.cs.uml.edu/~heines/teaching.html.

2.2 The General Music Methods II Course

The music course was the second semester of a two-semester sequence for aspiring music educators. The first semester provides an overview of the educational, philosophical, and psychological rationales that support varied musical experiences. Through exploration, cooperative learning, and divergent and analytical thinking, students establish a foundation for all levels of music learning consistent with the Massachusetts Arts Curriculum Framework. The second semester investigates the latest research on brain-based learning and is designed to involve students as active participants. The approach is multi-disciplinary within the field of music, employing improvisation, composition, performance, and listening exercises. All aspects of these courses are conducted in the same inductive manner advocated for the classroom. There are regularly scheduled fieldwork experiences at the Middle and High School level.

3. THE FOUND INSTRUMENTS PROJECT

The Found Instruments Project was conceived by the Music professor to get music education students to think about the learning of a new symbol system from the perspective of a total novice. She notes, “for many students, learning musical notation is an impediment to music making” [7]. A student in her class observed that “for students to truly comprehend standard musical notation, they must first create their own. Through that creativity and exploration, they will make the connections necessary to bridge the gap between their own creations and standardized music notation.”

3.1 Assignment for Music Students

Music students are assigned to find typical household objects that can produce several pitches or timbres. They are asked to design musical instruments from these objects and create compositions for them that adhere to common musical forms. Once their compositions are crafted, students devise notation systems that others can understand well enough to perform those compositions with little or no verbal or written direction. Their notation systems are not to resemble standard musical notation in any way, shape, or form. Students then present their instruments and compositions to classmates to explore and perform.

Figure 1 shows an excerpt from a typical student-created notation. This notation is for playing a pair of high-heeled shoes, which we see a student attempting to do in Figure 2. Note, however, that this is not a Music student, he’s a computer science student. Thus, it can be seen that the CS students got engaged with the music part of the project beyond the computer programming part described below. Likewise, as discussed later, the Music students got engaged with the computer part far beyond the music part.

3.2 Assignment for CS Students

Once the Music students had finished creating their notations and explaining them to the CS students, each paired up with a CS student who would create a computer program that allowed users to write music using that notation. These programs were modeled after Finale Notepad and Sibelius. It is important to keep in mind that the CS students had no prior musical experience, which caused some very interesting discussion of user interface issues.

Figures 3 through 5 give a feel for the interdisciplinary nature of the project. Figure 3 shows another notation created by a Music student, this time for playing a steam iron. Figure 4 shows the complementary computer program created by a CS student. And to complete the loop, Figure 5 shows two Music and one CS student collaborating on the program design. There were actually three instances in which the CS students presented their programs to Music students in rudimentary usability tests, each time observing and receiving feedback to improve their programs.

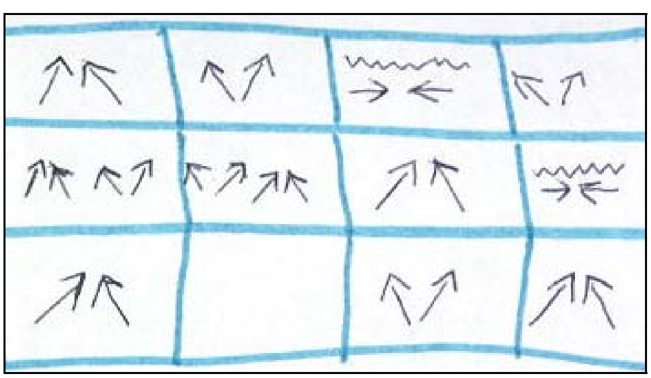

Figure 1. Kristen’s notation for playing a pair of shoes.

Figure 2. CS student Chase attempting to “play” a pair of shoes given Music student Kristen’s notation.

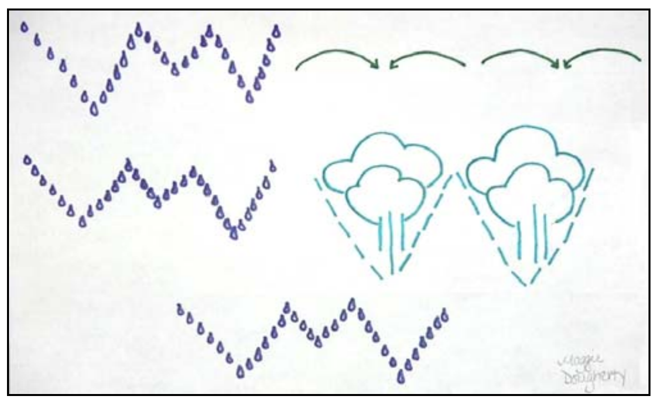

Figure 3. Maggie’s notation for playing a steam iron.

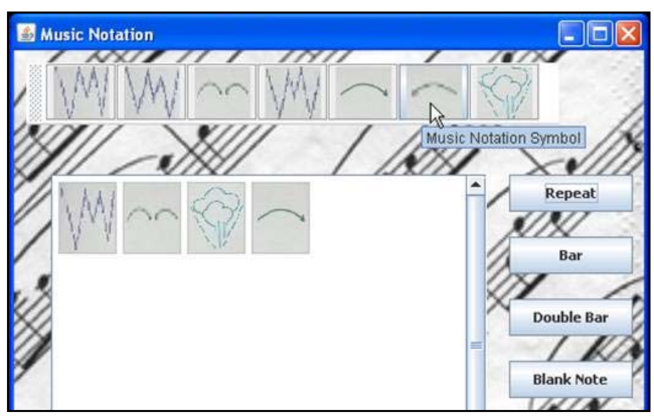

Figure 4. CS student Sophanna’s computer implementation of Music student Maggie’s steam iron notation.

Note: Videos of various Found Instruments presentations by both Music and CS students can be found at www.youtube.com/performamatics.

Figure 5. Maggie, Mike, and Sophanna reviewing Sophanna's computer implementation of Maggie's steam iron notation.

4. OBSERVATIONS

4.1 By the CS Professor

This is the second class in which the CS professor has used the "synchronized" class model with his GUI programming class. The first was with Prof. Jim Jeffers in a Web Art & Design class the previous semester, described in detail in Heines, Jeffers, and Kuhn (2008) [9]. In the current class, the professors attempted to correct some of the problems that were observed in that first class.

- The Found Instruments project was formally introduced to both sets of students simultaneously in a joint class.

  Prior to this class, the Music professor simply instructed her students to "bring in one or more household objects able to make a minimum of three different sounds." The CS professor shared these instructions with his students, and to the professors' surprise and delight, the CS students showed up at the class with "found instruments" as well. This immediately set a cooperative tone that lasted throughout the life of the project.

- Joint classes were structured with specific objectives.

  The first time we tried the synchronized course model, students sometimes floundered in the joint classes, wondering what they were supposed to do. This time, tasks were more clearly defined, such as who was to present what, or who was to review what, so that time was used more wisely and productively, which was more satisfying to everyone involved.

- Assignments were given shorter time frames to focus the work more clearly and thwart procrastination.

  As a matter of fact, the entire project was completed in the first half the semester, as opposed to spanning the entire semester as had been done in the previous synchronized class.

- The work done by each set of students was graded.

  In the previous class, only the CS students' work was graded. For the art students, the work was an add-on to the many other projects they were required to complete for the course.

As a result, the CS students commented that the "Music students were more interested in the outcome [than the art students in the previous semester] and more excited about what we did with their notations." The CS students also "enjoyed being exposed to music concepts that they wouldn't have any insight into."

4.2 By the Music Professor

The following comment by one of the Music students regarding their collaboration with the CS students is fairly representative of the many benefits of this interdisciplinary approach, "I love hearing different perspectives from people in totally different areas of study. It's really easy to forget what it was like to not be a musician and how you would have thought about music back then." It is our belief that as the work force is relying increasingly on multidisciplinary solutions to ever more complex issues, opening our students up to new ways of thinking and putting them in situations where they are once again novices will increase their sensitivities to what it's like to approach a problem from the viewpoint of a "beginner."

As observed by one of the Music students toward the end of the project, "It was interesting to see the different directions taken by the CS students. Those programs [that used drag-and-drop] tended to work well … The student who used the color flashes may not have understood the way or the reason traditional notation is used. It was more of a light show and could add to music, but would not serve a performer to recreate the music since there is no way to look ahead a measure or two." Another student stated, "I think it's been a really interesting experience. I think it's great that we got to try out our notation with them, considering most of them have had no previous musical training … I'm looking forward to seeing their abilities."

It should be noted that as this project was nearing completion, the Music students were coming up with ideas for incorporating another project with the CS students into our coursework. This represented a complete turnaround in their attitudes, since they began our collaboration with a great deal of skepticism. As suggested by Seifert & Mandzuk [14], "…learning communities do not happen automatically." In our case the shared planning and class times was instrumental in helping our students bond and take ownership of the group experience.

5. ASSESSMENT

The Performamatics approach, though only a year old, has yielded generally positive reactions from students and faculty.

5.1 Student Experiences

In group interviews with the evaluator, both CS and Music students expressed great enthusiasm for the collaboration and cited many benefits.

- The "real world" character of the project experience.

  CS students commented that when they leave school they will be working in interdisciplinary teams, and that the experience of collaborating with Music students gave them one of the few opportunities in their curriculum to experience this sort of work. Students in the previous semester's Performamatics course made the same observation and felt it was a great strength of the approach.

- The chance to experience other perspectives.

  For the CS students, the interaction with Music students gave them a taste of what it is like to meet with and learn from users. A Music student described a design conversation with his CS counterpart: "He said, 'We could have ten minutes of music and then you could erase everything.' I said, 'Do you know what a musician would do to you if [he] just wrote ten minutes of music and then [the program] erased everything? [He] would KILL you!'"

  The Music students, who will be teaching music to novice learners, also got to experience working with non-musicians. Some of the CS students made the sequence of notations run vertically rather than left to right. This caused the Music students to realize that left to right—like standard notation—is a convention that must be learned. This was one of many differences that caused students in both classes to reflect on things they ordinarily take for granted.

- The discovery of commonalities.

  As one Music student noted, he learned that "somebody else on campus [who] has nothing to do with us [that is, a CS student] has everything to do with us." While the differences between the groups, noted above, were useful pedagogically, the students also discovered how much they had in common, and how interesting it could be to explore each other's viewpoints. Another Music student said, "They are in a creative process just as much as we are when we create music… I saw a lot of similarities between what they were doing and what we were doing."

- The level of engagement.

  The fact that CS students showed up at their first meeting with the Music students bringing found instruments on their own initiative says much about their level of engagement. The engagement continued as CS students discovered their ability to communicate with the Music students and the intrinsic fun and interest of joint creation. One CS student commented, and the rest agreed, that having everyone in the class working on a different project was so much more interesting—and more instructive—than having every student working on the same assignment, as happened in many of their classes.

Students were nearly unanimous in feeling that the innovative collaborations introduced by Performamatics were valuable and should be retained and even expanded as part of the curriculum.

5.2 Faculty Experiences

Performamatics innovations have also had an impact on faculty. They give faculty a chance to include enhanced forms of active learning in their classes, with all the benefits discussed above. Joint classes with a faculty member in another discipline are also a learning experience for those professors and can break down the isolation that faculty sometimes feel. Course planning conversations and project meetings are opportunities for reflective practice and exploring ideas about pedagogy. NSF's CPATH program is aimed at revitalizing computer science, and the CS faculty members involved have felt that they themselves were revitalized as educators. This is clearly a key to transformative change in CS education—the transformation of the faculty.

Also, the "synchronized" classes model, in which an art or music class is scheduled to meet at the same time as a computer science class, but not necessarily in the same room, is an effective "low overhead" model for multidisciplinary learning. Performamatics classes in both semesters have used this model, avoiding the bureaucratic problems of having a new course approved, allocating student FTEs to faculty, and so forth.

While the model has limitations in that the students do not spend the full semester together, it also has advantages, particularly in more advanced courses where some measure of disciplinary content must be communicated. For example, The CS professor said that if the CS and Music students had been in the same room for every class he would not have been able to teach the necessary Java skills necessary to implement the music notation programs. He was able to do this because he had considerable time alone with the CS students. Students retain their disciplinary identity more fully in a synchronized than in a joint class, which can be a benefit for advanced students.

6. FUTURE DIRECTIONS

Given the positive reception—by students in both classes—of the Found Instruments project and the concepts that surround it, the professors have proposed to build an entire course around this project and its offshoots. In addition, they have proposed to make the new course truly interdisciplinary so that students from both the arts and the sciences enroll in the same class. By giving a single class two course numbers, one for arts students and one for sciences students, the arts students could use this class to fulfill part of their "science and technology" general education (GenEd) requirement, while the science students could use it to fulfill part of their "arts and humanities" GenEd requirement. Such an arrangement has been shown to greatly increase enrollments, since the course "counts" for something more than just a "free elective."

Although this arrangement may appear to reduce the impact of general education distribution requirements by giving students "half a course" outside their field, we believe that teaching science and the arts in a way that allows students to see the connection to their major interest is more valuable and will have a more lasting impact than an isolated standalone course.

The expanded course would go deeper into alternative music and graphic notation, perhaps drawing on the work of late 20th century composers such as John Cage, Helmut Lachenmann, George Crumb, Karlheinz Stockhausen, Gyorgi Ligeti, Krzysztof Penderecki, and Aaron Cassidy to name just a few, who use highly complex notation strategies to represent music and sounds to suggest what they want performers to do. For example, Figure 6 shows a sample of Cassidy's notation for a piece for solo trumpet [3]. We will explore how composers portrayed their ideas dating from the Renaissance up through contemporary composers and ways in which this form of notation became more popular as the aural palette expanded beyond traditional instruments into environmental sounds and found objects.

This course will allow CS majors to see the practical applications of computer science while introducing them to the basic fundamentals of music. It will expand their understanding of audio issues and the human factors of end user issues in multimedia software. The arts students will benefit from an understanding of the basic principles that underlie much of the technology in current use today. In addition, the interdisciplinary focus of this course will be of value to all students interested in working with and understanding some of the multimedia development software currently on the market through hands-on experiences.

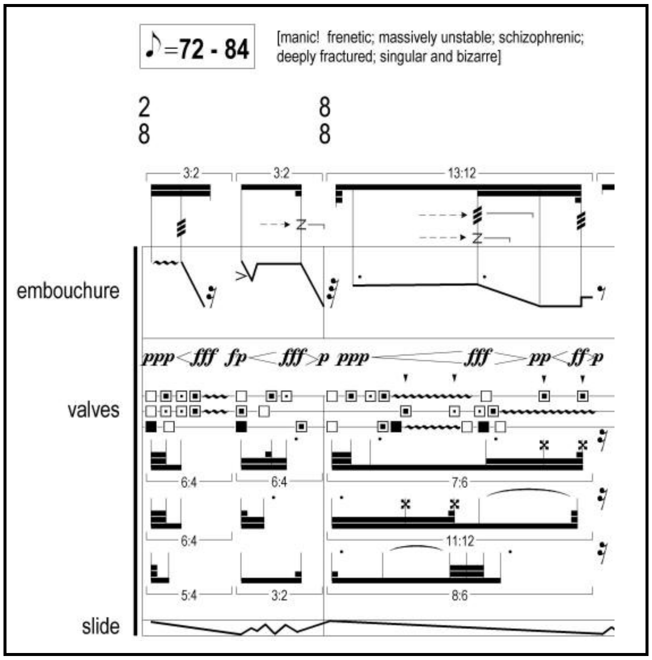

Figure 6. Notation excerpt from *Strange Smile* for solo trumpet by Aaron Cassidy [3].

7. ACKNOWLEDGMENTS

The work described in this paper is based upon work supported by the National Science Foundation under Grant No. 0722161, “CPATH CB: Performamatics: Connecting Computer Science to the Performing, Fine, and Design Arts.” *Principal Investigator:* Jesse M. Heines. *Co-Principal Investigators:* Fred G. Martin, Gena Greher, Jim Jeffers, and Karen Roehr. *Senior Personnel:* Sarah Kuhn and Nancy Selleck. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

We gratefully acknowledge the contributions of the Music and CS students who participated enthusiastically in this study and made it so enjoyable for faculty and students alike. (Written releases to reproduce photographs of students and screen captures of their work have been obtained.) We also acknowledge the contributions of Research Assistant Daniel Gabriel to this paper.

Additional information on the Performamatics project can be found at www.performamatics.org.

8. REFERENCES

[1] Campus Technology (2007). *Microsoft: Time to 'Panic' in Search for CompSci Workers*. campustechnology.com/news_article.asp?id=19886&typeid=150 *accessed* January 5, 2007.

[2] Carter, J. (2001). *Arresting the Decline: conversations with female CS undergraduates*. Computing Laboratory, Univ. of Kent at Canterbury. www.cs.kent.ac.uk/pubs/2001/1218 *accessed* January 21, 2007.

[3] Cassidy, A. (2007). *What then renders these forces visible is a strange smile (or, First Study for Figures at the Base of a Crucifixion)*. www.aaroncassidy.com/music/smile.htm *accessed* August 25, 2008.

[4] Foster, A.A. (2005). *Student Interest in Computer Science Plummets.* Chron. of Higher Ed. **51**(38):A31.

[5] Geigel, J., & Schweppe, M. (2005). *Virtual theatre: a collaborative curriculum for artists and technologists*. Proc. of the ACM SIGGRAPH 2005 Educators Program, Article No. 16. Los Angeles, CA.

[6] Gestwicki, P., Sun, F.-S., & Dean, B. (2008). *Teaching game design and game programming through interdisciplinary courses.* Journal of Computing Sciences in Colleges **24**(1):110-115.

[7] Greher, G., & Heines, J. (2008). *Interdisciplinary Collaboration: Connection Computer Science and Music Students to the Benefit of Both.* Assoc. for Technology in Music Instruction. Atlanta, GA.

[8] Heines, J., & Schedlbauer, M. (2007). *Teaching Object-Oriented Concepts Through GUI Programming*. Eleventh Workshop on Pedagogies and Tools for the Teaching and Learning of Object Oriented Concepts, held at the 21st European Conf. on Object-Oriented Programming. Berlin, Germany.

[9] Heines, J., Jeffers, J., & Kuhn, S. (2008). *Performamatics: Experiences With Connecting a Computer Science Course to a Design Arts Course.* Int'l. Jrnl. of Learning **15**(2):9-16.

[10] Kim, H.J., Coluntino, D., Martin, F.G., Silka, L., & Yanco, H.A. (2007). *Artbotics: Community-based collaborative art and technology education*. ACM SIGGRAPH Educators Program. San Diego, CA.

[11] Klawe, M., & Shneiderman, B. (2005). *Crisis & opportunity in CS.* Comm. of the ACM **48**(11):27-28.

[12] McDonald, J., & Wolfe, R. (2008). *Using computer graphics to foster interdisciplinary collaboration in capstone courses.* Jrnl. of Computing Sciences in Colleges **24**(1):83-90.

[13] Round, A. (2006). *The Decline in Student Applications to Computer Science and IT Degree Courses in UK Universities*. Computing Research Assoc. Conf. Proc. Snowbird, UT: ACM.

[14] Seifert, K.L., & Mandzuk, D. (2006). *Student Cohorts in Teacher Education: Support Groups or Intellectual Communities?* Teachers College Record **108**(7):1296-1320.

[15] Yanco, H.A., Kim, H.J., Martin, F., & Silka, L. (2007). *Artbotics: Combining Art and Robotics to Broaden Participation in Computing*. Proc. of the AAAI Spring Symposium on Robots & Robot Venues. Stanford Univ., CA.