# Explorations in Musical Tinkering

Eric Rosenbaum

A.B. in Psychology and Mind/Brain/Behavior, Harvard University, 2002
M.Ed. in Technology in Education, Harvard University, 2003

Submitted to the program in Media Arts and Sciences,
School of Architecture and Planning,
in partial fulfillment of the requirements for the degree of
Doctor of Philosophy in Media Arts and Sciences
at the Massachusetts Institute of Technology
February, 2015
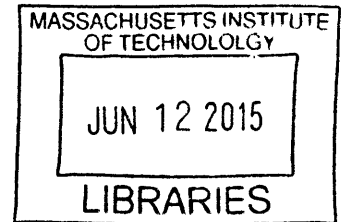
Signature redacted

Author   Eric Rosenbaum
Program in Media Arts and Sciences
January 2015

Signature redacted

Certified by   Mitchel Resnick
LEGO Papert Professor of Learning Research
Program in Media Arts and Sciences

Signature redacted

Accepted by   Pattie Maes
Associate Academic Head
Program in Media Arts and Sciences

1

Explorations in Musical Tinkering

Eric Rosenbaum

ABSTRACT

This thesis introduces the idea of "musical tinkering," exploring how to engage people in playfully creating and experimenting with their own musical instruments and compositions. I discuss the design and study of two new tools for musical tinkering, MelodyMorph and MaKey MaKey. MelodyMorph is an iPad app for making musical compositions on the screen. MaKey MaKey is an invention kit that lets you transform everyday objects into physical-digital musical instruments.

Two themes of musical tinkering, the loop and the map, are woven throughout this thesis. Loops are feedback processes. They range in scope from rapid iterative design, through interpersonal interaction and creative emergence, to longer-term personal transformation. Maps are active visualizations. We use them to externalize our thought processes, and we fluidly manipulate them as we tinker, linking graphical or tangible symbols to musical sounds. I use loops and maps as the basis for design concepts for musical tinkering tools and to analyze musical tinkering as a learning process.

I present case studies of middle school students tinkering as they use MelodyMorph to compose musical stories, reconstruct tunes from video games, and make musical cartoons. I also present case studies of MaKey MaKey, showing how people have used it to tinker with music "in the wild," in my own workshops, and in the work of other educators. Through these case studies I characterize musical tinkering using the concepts of musical landscape-making, musical backtalk, and musical inquiry. I show that loops and maps intertwine in the processes of collaborative emergence, inventing new maps for new instruments, and tinkering with musical ideas and musical attitudes. Finally, I conclude with visions for remaking the landscape of musical tinkering in the future.

Explorations in Musical Tinkering

Eric Rosenbaum

Signature redacted

Thesis Reader   Edith Ackermann
                       Visiting Scientist, MIT Media Lab
                       Senior Research Associate, Harvard Graduate School of Design

Explorations in Musical Tinkering

Eric Rosenbaum

Signature redacted

Thesis Reader

Tod Machover
Muriel R. Cooper Professor of Music and Media
MIT Media Lab

Explorations in Musical Tinkering

Eric Rosenbaum

Signature redacted

Thesis Reader   Rena Upitis
Professor of Arts Education
Queen's University

Acknowledgements

Table of Contents

Chapter 1:
Musical Tinkering with Loops and Maps

Sandy and Daniel, 10 years old, are using an iPad app called MelodyMorph to create a musical comic strip. Each boy has drawn a character on the screen, and instead of writing dialog, they put melodies into the speech balloons. Daniel drags several colorful circles from a palette into his character's speech balloon, each circle representing a piano note. He taps the circles one at a time to play a short melody, rearranges the melody and plays it again, forwards and backwards. He seems unsatisfied, and agrees with Sandy that something is missing. He searches the palette until he finds a note that sounds right for the ending. Finally he records his melody, encapsulating it in a new object that he can tap to play the whole thing. Later, Sandy plays the entire musical comical strip in sequence, narrating the story along with the music: at first Daniel's character is sad, but when Sandy's character offers to play a new video game with him, he responds happily.

A college-aged woman named Ruth is part of a team building an interactive musical installation with helium balloons. They are using an invention kit called MaKey MaKey, which can trigger a sound on a computer when a connection is made and a circuit is completed. The team quickly finds that latex balloons are not electrically conductive, and they begin to experiment with foil and wires. But these are too heavy and make the balloons sink. Ruth creates a new lightweight connection, by painting spirals of conductive paint onto the tops of the balloons. Now they can attach sheets of aluminum foil to the ceiling, and each time a balloon touches the ceiling, it plays a note. But what sound should a Balloonophone make? Do the pitches of the notes relate to the color of the balloon? They find that the balloons don't just play single notes, but bounce repeatedly against the foil, triggering a stuttering rhythm. The group debates: should they design a way to prevent this behavior, or instead embrace the music it makes?

These stories describe events from my workshops, which introduce tools for musical tinkering. The first story is about MelodyMorph, an iPad app for creating musical instruments, compositions, and stories on the screen. The second is about MaKey MaKey, an invention kit that lets you transform everyday objects into physical-digital musical instruments. In each, we see a process of musical tinkering.

Musical Tinkering

Here is my broad definition of musical tinkering:

Musical tinkering is a pathway into music-making through playful creation of musical artifacts.

I see musical tinkering as a "pathway into music-making" because I am especially concerned with beginners who might not otherwise have opportunities to create their own music. Experts and others who are already music creators can tinker with music as well, finding pathways into unexplored or challenging areas. But the focus for the design of the tools in this dissertation is on empowering creative experiences for beginners.

The process of "playful creation" is at the heart of musical tinkering. Playful creation can mean embracing the unexpected, the humorous, and the ridiculous. It can mean testing the boundaries, to find out what's possible. It can mean experimenting without quite knowing what you're trying to make. But it always involving making things.

By "musical artifacts" I mean physical or digital creations that persist over time. This means that musical tinkering does not include musical activities such as performing composed or improvised music, or exploring an existing musical interface. Instead, it involves the creation of musical instruments, musical compositions, and interactive artifacts that blur the boundaries between them.

Leitmotifs: the Loop and the Map

I will frame my analysis of musical tinkering using a musical metaphor, the leitmotif.

A leitmotif is a theme, often a short melody, that recurs throughout a musical composition. The composer states the leitmotif simply at the beginning and weaves it through the piece as the composition grows and changes.

In my explorations of musical tinkering two images have emerged: the loop and the map. These are my leitmotifs of musical tinkering.

A loop is a process in time.

A map is a shape in space.

A loop is feedback. We act, and the world brings information back, so that we can learn and try again.

A map is a model. We make a picture so that we can see the world, and our own thoughts, in a new way.
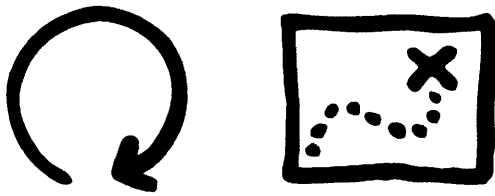
Figure 1: A loop and a map

Where are the loops and maps in musical tinkering? We can see some hints in the anecdotes above. Daniel tinkers with melodies—composing, listening, and composing—in a loop. The arrangement he makes on the screen helps him see his composition as he transforms it, serving as a map. Ruth tinkers with materials and sounds, her iterative experiments forming a loop. The balloons become an instrument, their arrangement in space determining the landscape of possible melodies they can make, like a map.

Tinkering Loops

The context for my research begins with a philosophy of learning called constructionism (e.g. Papert, 1993). In a constructionist learning environment, people learn through making things that are personally and socially meaningful. Tinkering, and loops, are an integral part of this process. One way to tell the story of constructionism is to begin with the cybernetic turtle, a creature made of loops.

Cybernetics is a study of control systems.The field of cybernetics introduced a view of the world as full of feedback loops (Wiener, 1948). A machine modifies its environment, senses the resulting changes, and uses this information to make new modifications. An organism acts on the world, perceives the resulting differences, and uses that information to act again. This is a kind of causality in spirals, or loops. Grey Walter's cybernetic turtles (Walter, 1950) were the world's first autonomous robots, using feedback to control their movement. They were simple electronic creatures on wheels that could move towards a light: sensing greater brightness to one side, turning and moving toward it, sensing a new difference, and moving again. The cybernetic turtles showed how a machine could act like an organism, using feedback loops to generate life-like behaviors that reduced the difference between their current state and a goal.
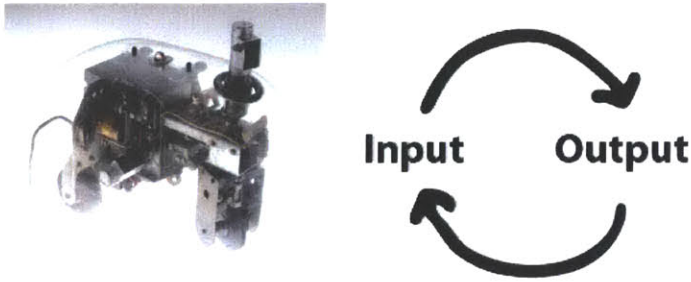
Figure 2: On the left, a cybernetic turtle like ones invented by Grey Walter. It has a light sensor, a steering mechanism, and a transparent shell with a bump sensor. On the right, a schematic for a simple feedback loop.

In the 1960s and 1970s, Seymour Papert and his group at the MIT AI Lab created a new kind of robotic turtle. This one was not autonomous, but designed to be programmed by children. Using a new programming language called Logo, children could issue commands to the turtle to make it move and turn on the floor. By grouping these commands into programs, they could make the turtle execute a complex sequence of movements. With a pen dragging along underneath, the turtle could make drawings.
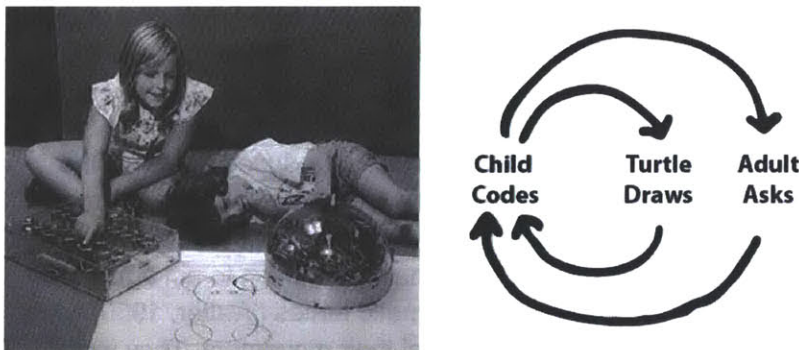
Figure 3: On the left, a child programs a robotic turtle that draws with a pen as it moves. On the right, a schematic representing looping feedback between the child and the turtle, and between the child and the adult.

Now, the feedback loop has become larger than the robot. The system includes not just the turtle, but the child as well. The child writes a program in the hopes of making the turtle draw a house and watches the turtle run the program. The roof is upside down, so she makes a change to the program and tries again. The feedback loop might grow again: an adult asks her questions about her program, or makes a comment on the drawing. The system as a whole now has behavior that is not simply directed toward a goal. Perhaps upon learning from an adult how to make the program repeat many times (another kind of loop, inside the turtle), she'll try copying the house around a circle, and start transforming it into a drawing of a

flower. The system is partly governed by the child's changing goals and partly by the adult's, to help the child learn new ideas. As feedback systems grow in complexity, unexpected new phenomena can emerge.

Mitch Resnick's creative learning spiral (Resnick, 2007) is a more recent development in constructionism, emphasizing creative thinking in the learning process. The creative learning spiral consists of imagining, creating, playing, sharing, and reflecting. People imagine what they want to make, create something, play with what they made, share it with others, and reflect on the experience. But the spiral is really a loop. Reflecting leads back to imagining, starting the cycle again.
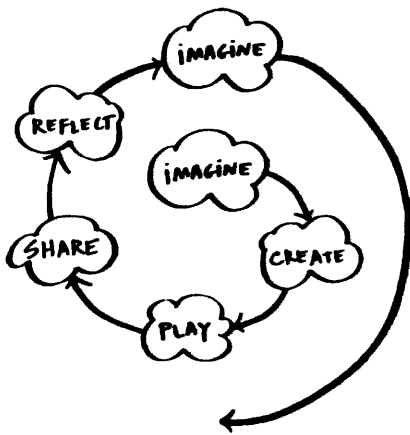
Figure 4: Resnick's creative learning spiral

As we enrich our view of loops in tinkering, we can compare the different kinds of loops. There are quick loops such as reprogramming a turtle or rearranging a melody. There are slower loops, in which a person moves through the creative learning spiral: building a new kind of musical instrument, trying out it with an audience, and refining it. There are also loops that involve interactions between several people who influence each other's ideas.

Don Schön's ideas about reflection can help to refine this view of loops. He differentiates between "reflection-in-action" and "reflection-on-action" (Schön, 1983). In his view, reflection-in-action is responding in the moment to a situation, using intuition and knowledge of prior experiences to act immediately. Reflection-on-action, by contrast, happens after and outside the situation, in the analysis of the actions and their outcome. These processes can be described as loops over different time scales. They each contain the idea of the feedback loop in which information coming back from one action feeds into a decision about the next. But reflection-in-action occurs over seconds or minutes, and reflection-on-action happens over hours or days or weeks or longer.

Reflection-In-Action   Reflection-On-Action
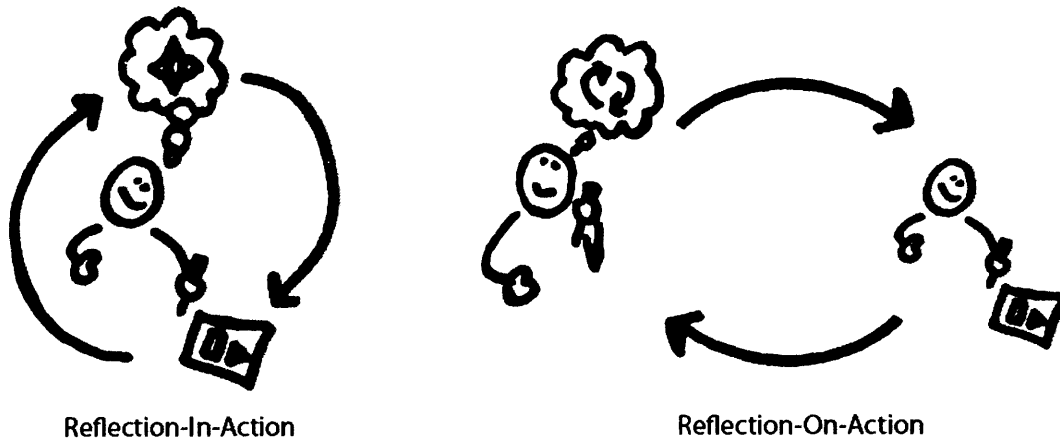
Figure 5: On the left, reflection-in-action, in which a person imagines possible drawings as they draw. On the right, reflection-on-action, in which a person thinks about their own drawing process, in between drawings.

Schön writes about the "reflective conversation with the material of a design situation" (Schön, 1992), drawing a metaphor between verbal conversation between people and the interaction between a designer and their material, such as an architectural drawing. One type of conversation he describes is "seeing-moving-seeing." This is a feedback loop that unfolds, for example, as a designer looks at her drawing, notices a problem, makes a change to the drawing, and then looks again, seeking ways to improve it further. Sometimes by looking again she can see the situation in a new way. Perhaps a small change has had a surprisingly large effect on the situation as a whole, leading to a change in perspective. Schön refers to this unexpected feedback from the situation as "backtalk." This image of backtalk in a conversation with materials captures an essential aspect of tinkering loops: small iterative experiments can lead to new insights. These insights can inspire experiments that lead in a new direction.

Schön has also written about a larger system of "double loop learning" (Argyris and Schön, 1978), connecting loops at multiple scales. Where the smaller loop involves work toward a particular goal, the larger loop around it is the process of questioning and changing the goal itself or the framing situation around it. More generally, we know that the loops in tinkering are not single or isolated. There is always interaction between interconnected loops. The creative learning spiral, for example, is an idealization of a process that involves many crossings back and forth between phases. We know that people do not simply progress forward, from imagining to creating to playing to sharing to reflecting. Instead we see them jump, for example, from sharing back to playing, or skip from playing to reflecting, or repeatedly imagine and create.

Double Loop Learning
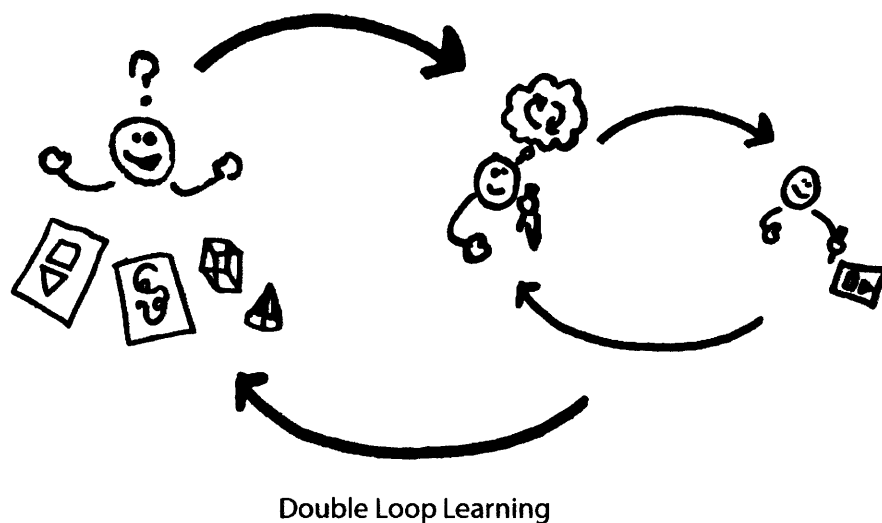
Figure 6: Double-loop learning. In the inner loop (on the right), the person draws, and reflects on their drawing process. In the outer loop, the person questions how and what they should be drawing.
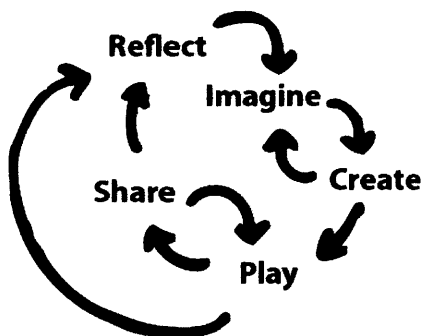
Figure 7: Based on Resnick's creative learning spiral, a complex creative learning loop made up of one possible arrangement of smaller loops.

The more traditional research on learning explores the role of feedback in learning in the form of self-regulation and co-regulation. Butler and Winne (1995), for example, synthesize research on feedback and self-regulated learning. The findings they review show that the most effective learners regulate their own learning process in several ways. These learners set their own goals and monitor their progress toward them, modify the goals as needed, and manage their motivation. The authors also suggest that a more fine-grained analysis of the role feedback in learning is needed. The studies typically look at its effects in aggregate for an entire task, obscuring the moment-to-moment dynamics of feedback. The image I use in this

dissertation, of learning through tinkering in loops across time scales, leaves room for description of these rapid feedback processes.

The research on co-regulated learning focuses on interpersonal feedback, focusing on the loops at intermediate time scales. For example, Azevedo et al (2003) found that students perform better when they study with a tutor who helps them self-regulate than in conditions where they had written self-regulation prompts or none at all. The social aspects of loops in tinkering of course involve many factors that can help learners progress, including this type of cognitive self-regulation, as well as affective factors like motivational support, and the creative feedback included in the "sharing" stage of Resnick's creative learning spiral.

A full description of tinkering loops takes into account the many kinds of feedback loops. Loops range across time scales, from the small rapid moves in a conversation with the material, up to the longer, more deliberate loops of reflection-on-action, and the more radical outer loop of double-loop learning. Loops are complex, not constrained to simple cycles but instead forming interconnected networks with branching paths and smaller loops nested inside larger ones. And loops can be made up of different kinds of parts. A conversation with materials is between a person and an artifact, but the larger loops can incorporate other people, such as collaborators and teachers.

To recapitulate, tinkering loops are learning processes that we can divide into three categories: inner loops, context loops, and outer loops.

| Inner-Loop | Context-Loop | Outer-Loop |
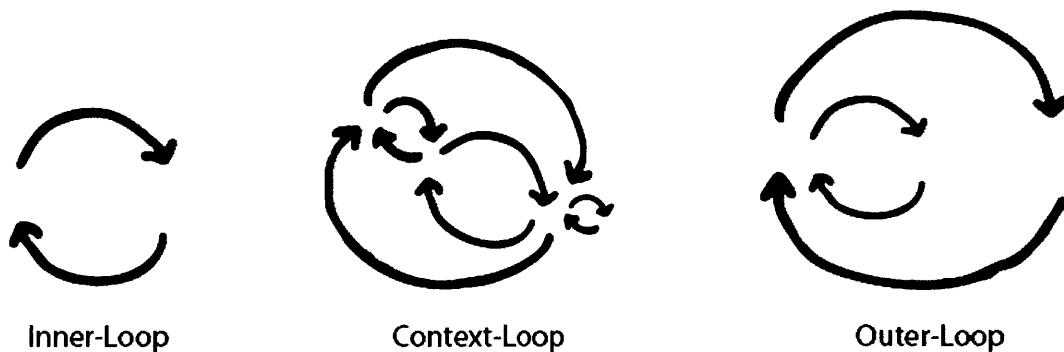
Figure 8: The three types of tinkering loops.

Inner-Loops are fast feedback. This is reflection-in-action, and conversation with materials.

Context-Loops are complex. They form branching networks of causality, often interpersonal.

Outer-Loops are slow and powerful. They involve transformations and questions.

Tinkering Maps

Maps provide another important context for my work. A map is a model of the world, an abstraction of the world's parts and their relationships. A map can also be a model of our inner world, a diagram of our thoughts. When we tinker, we think with our hands, leaving traces of our thinking in the material with which we tinker. These traces can form a kind of map of our ideas. If this map is modifiable, we can use it to help modify our own thinking (in a loop). The map is even more powerful if it can serve not only as a diagram, organizing symbols, but also as an actor, causing events in the world. I will present the concept of tinkering maps as maps that externalize our ideas, that can be transformed, and that can act as links, both symbolizing and acting.

How can maps externalize our ideas? A map, as we know, is "not the territory" (Korzybski, 1933). Maps are partial, distorted versions of the real world. A hand-sketched map of a route across town may alter distances, simplify paths, omit some landmarks, and exaggerate others (Tversky, 2002). But we sketch route maps this way because these properties reflect our mental models of space and navigation, and help us to communicate about them. The distortions in this kind of map make them truer to our mental maps. The structures in these sketches reveal something about the structures in our thinking.

Concept mapping is a strategy for explicitly representing the relationships between ideas (Novak, 1984). A concept map is a network of bubbles connected by lines. Each bubble contains an idea, in the form of a short description, and the lines connecting them represent propositional relationships between them, such as membership in a category (e.g. "dog" connected to "mammal" connected to "animal"). Proponents of concept maps suggest that they are effective learning tools because their interconnected structure reflects the actual structure of knowledge in our minds. This may or may not be so, and is certainly a simplification. By examining tinkering maps, we may see a variety of ways in which people use external structures to reflect the organization of ideas in their minds.

Most maps are static, but tinkering maps can be modified as we tinker. In the stories of Sandy and Ruth, the graphical objects representing notes on the iPad screen and the floating musical balloons can be endlessly rearranged and transformed.

Maps are also normally limited to passive description. But what if you could use a map to make things happen? The components of a tinkering map can link symbol and action. The graphical note objects and the musical balloons are each symbols, denoting a particular musical pitch. But they also act like triggers. When you touch them, they make a computer play that particular musical note. In this way, tinkering maps combine the qualities of diagrams with those of user interfaces.

We don't necessarily need technology to create something that acts both as a symbol and as a trigger for an event. Human language plays this dual role and perhaps underlies our ability

to recreate it with other systems. We use spoken language, beyond making statements that refer to things in the world, to actually make things happen. The philosopher J. L. Austin wrote about the ways in which spoken words can themselves be actions with external effects (Austin, 1975). He refers to these as "perlocutionary" speech acts. For example, we ask questions in order to elicit answers, or make requests or commands in hopes of modifying the behavior of other people.

Technology amplifies this ability of human language to effect change. Programming languages are a newer form of language which has even greater power to both symbolize and to act. In "The Structure and Interpretation of Computer Programs" (Abelson and Sussman, 1984), the authors use the metaphor of a magic spell: "People create programs to direct processes. In effect, we conjure the spirits of the computer with our spells."

Physical objects can also simultaneously be symbols and tools. "Perceived affordances" (Norman, 1999) are the actions that a user, seeing a particular object, believes that object makes it possible for them to perform. For example, seeing the handle on a door, you assume that it is possible to pull on the handle to open it. Seeing an icon on a screen, you believe, based on prior experience, that you can click on it to launch an app or open a file. Features that lead to perceived affordances, like the parts of a tinkering map, occupy the intersection of abstract symbols and behaving objects.

Because tinkering maps are manipulable, they have what we might call "manipulable affordances." In addition to having parts that can both symbolize and act, tinkering maps are also flexible and changeable. Computer programs are an especially good example of this flexibility. Some languages, such as Smalltalk-80 (Goldberg, 1989), allow words, as variable names, to refer to almost anything else in the language. Almost any word can be redefined to a new meaning, with a new effect.

Logo programs have all of the properties of tinkering maps. Seymour Papert envisioned Logo as a way to learn by thinking about thinking. A Logo program can resemble a thought process, for example, when it reflects the programmer's thinking about how to make the turtle move. The programmer can modify the program, to reflect changes in her thinking. Words in the program symbolize properties and actions, but can also cause the actions to occur when the program is run.

To recapitulate, a tinkering map is a special kind of map that can be used for three kinds of activity: externalizing, transforming, and linking.

Externalizing makes ideas and thought processes visible.

Transforming changes and re-shapes the map.

Linking simultaneously refers to something and makes it happen: symbolizing and acting.

| Externalize | Transform | Link |
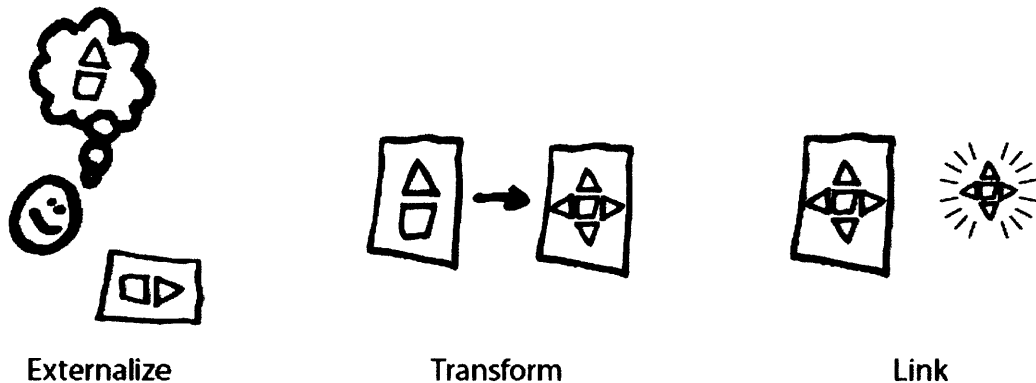
Figure 9: The three activities of tinkering maps.

Musical Tinkering

What do tinkering loops and tinkering maps look like when we are tinkering with music? I will raise some questions here, to be explored in later chapters.

Loops in musical tinkering

The quickest of the tinkering loops in music, the inner loop, is like Schön's "conversation with the materials of a design situation." How does a conversation with musical materials unfold over a short time scale? What is the musical form of "backtalk?"

These rapid loops in musical tinkering rely on feedback through listening, but they also engage the body. The Dalcroze tradition of music education focuses on embodiment, taking physical movement to be the basis for musical expression (e.g. Seitz, 2005). Musical tinkering with physical materials combines expressive musical gestures with hands-on making, combining two forms of embodied cognition. For example, as learners experiment with connecting balloons to a circuit to trigger sounds, they are thinking with their hands in two ways. They are tinkering with both the physical properties of the materials, rapidly testing connections in different ways, and with the musical properties of the balloons as an instrument, trying out different arm and hand movements as they move the balloons and listen to the results. How will these two forms of embodied cognition interact in musical tinkering? Will intuitive musical gestures generate ideas for physical construction? Will the gestures of physical making inspire expressive musical movement?

Ackermann (2005) analyzes feedback processes in play, which are similar to the inner loops of tinkering. She looks at toys that balance responsiveness and autonomy, which she calls AniMates. One example is a kite, which you control by pulling a string, but which also exerts its own force, pulling with the wind. In the example of the balloonophone, the balloons bounced against the ceiling in a way that made them challenging to control. What happens when musical materials have their own autonomy?

I will also look at loops over longer time scales in musical tinkering. Bamberger's (2013) work provides one view. She investigates both reflection-in-action and reflection-on-action, as children develop their own notation for simple clapped rhythms. For example, she observes a moment of reflection-on-action as a child works on creating their own notation for a clapped rhythm. Only on reflection, stepping outside the activity, can the child extract from the rhythm the underlying beat, which structures time without always being heard. What new understandings will be revealed by reflection-on-action, in musical tinkering?

In a discussion of theater improvisation, Sawyer (2002) describes the dynamics of improvised dialogue as an interplay between two processes, forming a complex feedback system. In "collaborative emergence," the performers use accumulating details to build up a story together. In "downward causality," the emerging context of the shared story constrains the narrative moves that the improvisers can make, which in turn helps them to build up the story. This same interplay might be used to describe the emergent structure of an improvised performance by a jazz group, in which musical gestures both build up and constrain the emerging musical context. Will we see similar dynamics in collaborative musical tinkering processes? How do these dynamics change when the group is collaborating not on the performance of a continuously flowing musical composition, but on the construction of an interactive musical artifact?

Finally, what evidence will we see of the longest loops in musical tinkering, the outer-loops of questioning and transformation? Dillon (2007) shows how meaningful engagement with music can bring about personal, social, and cultural transformation. Will musical tinkering activities support these transformations? Can musical tinkering lead learners to ask the re-framing questions of double-loop learning? Can it empower them, transforming their attitudes about who they are, and their role as creators of musical culture?

The three types of tinkering loops will guide some of the questions I ask in this thesis. As I reflect on learners' experiences with MelodyMorph and MaKey MaKey, I will seek to understand how the inner, context, and outer loops of musical tinkering evolve.

Maps in musical tinkering

I have described three activities that characterize the use of tinkering maps: externalizing, transforming, and linking. How will these activities work in musical tinkering maps?

The idea of a map is deeply woven into our (current Western) conception of musical structures. Actual musical experiences continuously flow and change over time, but our musical culture has created systems to freeze it in place, using standard music notation and other visual representations. These are externalized musical maps. But what do our internal musical maps look like? Marvin Minsky (1982) has suggested, by analogy to the way we build up a mental map of a physical space as we explore it visually, that we build up maps of musical structure in our minds as we listen:

> "How do both music and vision build things in our minds? Eye motions show us real objects; phrases show us musical objects. We "learn" a room with bodily motions; large musical sections show us musical "places." Walks and climbs move us from room to room; so do transitions between musical sections. Looking back in vision is like recapitulation in music; both give us time, at certain points, to reconfirm or change our conceptions of the whole."

I suggest above that tinkering maps are external structures that contain traces of our mental maps. When we tinker with musical artifacts, the mental maps we use to structure these external maps will not just be organized like standard notation, in a score showing pitches over time. Will musical tinkering maps reflect the kinds of musical "objects" and "places" which Minsky describes? What do musical ideas look like, when they are externalized in a tinkering map?

Research on children's invented notation gives us some clues as to what these intuitive maps might look like. In Jeanne Bamberger's (2013) analysis of children's invented notations for rhythms, she notes a distinction between two complementary types, which she calls "figural" and "formal" drawings. The "figural" drawings have structures that reflect features like perceptual groupings of and boundaries between rhythmic events, without a quantitative representation of musical timing. The "formal" (or "metric") drawings have structures based on a consistent measurement of musical duration.
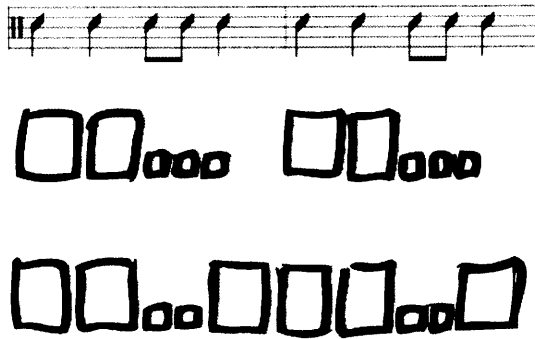
Figure 10: The same rhythm shown in three ways. Top: standard music notation. Middle: a figural notation, emphasizing the perceptual boundary between two phrases. Bottom: a formal notation, that accurately shows the durations but obscures the phrase boundary.

Bamberger points out that neither type is better or worse; while the formal drawings can capture rhythmic relationships with enough detail to reproduce the original rhythm, they can also obscure the gestalt features of our perception of rhythmic groupings that the "figural" drawings reveal. Based on these observations, we might ask how these figural and formal aspects will emerge in musical tinkering. Will we see both figural and formal properties in musical tinkering maps?

Bamberger (2013) also describes a distinction between two approaches to reconstructing melodies, which she calls path-making and map-making (note that Bamberger uses the term "map" in a different way than I do). She uses a task in which the child must recreate a known melody using a set of metal bells which play different pitches. Path-makers create a sequential arrangement of bells in the order of the melody. Map-makers, by contrast, first arrange their bells in order of pitch from lowest to highest, creating a musical scale, and then work out how to play the melody on this scale. The path-maker responds to the flow of musical events over time by creating a continuous sequence of bells that recreate the melody as she moves her hand along them. Her understanding of the melody is bound up with the process of moving along her path in time. Map-makers, on the other hand, can step away from their pathway in order to create a fixed spatial structure on which they then construct a path (in my scheme, "tinkering maps" can include both what Bamberger calls maps and what she calls paths). Bamberger notes that while she observes a progression in children from path-making to map-making strategies, map-making is not intrinsically better. Instead, the goal is to be able to use both as needed. In the broader world of musical tinkering, moving beyond tune reconstruction tasks into improvisation and composition, will we see a progression from path-making to map-making, or a mix of the strategies according to context? In other words, how will learners navigate and transform their musical tinkering maps?

Upitis (1991) takes a broader developmental approach, describing a sequence of stages in the development of children's invented notation of melodies. In the first stage, at the youngest ages, children draw pictures, responding to the music with gestures or images, but no musical symbols. In the second stage, they begin to show awareness of a "musical alphabet" but without definite use of musical symbols to denote pitch or duration, or the organization of these things into a sequence. In the third stage, children use separate symbols, but these may stand for individual notes, groups of notes, or entire musical phrases. In the fourth stage, a mix of musical features appears in the children's notations, as they begin to show awareness of rhythmic and melodic properties. Here, children begin to systematically represent features like melodic contour (the movement of a melody up and down in pitch) and musical phrase groupings. They are aware that a kind of musical grammar exists governing these properties but are not yet able to notate the properties in a way that consistently allows a reader to reproduce them. Finally, in a fifth stage, the notations have many of the properties of standard music notation, and it is possible to reconstruct accurate rhythmic structures and relative pitches. In musical tinkering, will we see the maps, as externalizations of musical thinking, follow a similar progression in sophistication?

Will storytelling play a role in the construction of musical tinkering maps? Blair (2007) worked with fifth-graders as they constructed graphical musical scores of existing pieces, which she calls "music maps." First the students listened to a piece, gesturing and drawing in the air along with it. Next they transformed their kinesthetic responses into a drawing of the melodies changing over time. This activity was based on their prior experience with similar teacher-created maps emphasizing melodic contour. The students collaboratively refined their maps and then "performed" them for the class, tracing the map with their hands or gesturing to it along with the music. She describes this map-making process as a way for the students to tell the story of their experience of listening. The creation and sharing of the maps make up a process of narrative inquiry, through which the students come to understand their own hearing and each other's hearings of a piece of music. In musical tinkering, the maps are of a different kind because they are playable representations of compositions. Will we see a similar kind of narrative inquiry unfold in their creation? Will the students externalize the stories in their minds as a way to structure their tinkering maps?

The musical maps discussed so far only capture musical properties like rhythm and pitch. Music is of course made up of numerous other qualities. We can use maps to show structure, such as repetition of themes or verses and choruses. We can illustrate expressive qualities as well, with continuous or descriptive notations denoting timbres or emotions. Musical maps can also incorporate lyrics and narrative or contextual elements. What mix of these features will appear in tinkering maps, as externalized traces of our inner musical worlds?

The transformability of musical tinkering maps creates some new possibilities. Digital technology allows us to set up entirely arbitrary relationships ("mappings") between input and output. Any visual representation or physical form can correspond to any sound. Prior to the existence of electronic instruments, the properties of physical materials created constraints on

these relationships. Acoustic properties, for example, govern the layout of holes on a woodwind instrument, and mechanical properties constrain their re-positioning as keys on instruments like the flute and clarinet. What new forms will we see, when we are totally freed from these constraints? Machover's (1992) Hyperinstruments project and its descendants use technology to extend musical instruments in several directions: transforming existing instruments and inventing entirely novel ones, as well as building systems both for virtuosic performers and for children and novices. Tinkerable instruments may open yet another space of possibilities, in which end-users can invent and creatively modify their own digital instruments. Will people recreate the familiar constraints of acoustic instruments? Will they invent new relationships and discover new kinds of constraints, as they transform their tinkering maps?

Finally, what are the implications of the fact that musical tinkering maps can link symbol and action? There is a whole literature of musical scores that move beyond standard notation, using a wide range of graphical notation and other visual imagery to express musical concepts (e.g. Sauer, 2009). These scores can be seen, perhaps, as the "invented notations" of modern composers. But generally these scores are still meant to be read by musicians who perform the piece on instruments, which is to say that the notation acts as a symbolic language. What changes when a score can function both as a set of symbols and as an interface for playing the music itself? How will we compose music using tinkerable maps? In this thesis I'll explore the three activities characteristic of tinkering maps: externalizing, transforming, and linking. As I reflect on learners' experiences with MelodyMorph and MaKey MaKey, I will investigate the role these activities play in musical tinkering.

The Big Picture

The remainder of this thesis focuses on two broad questions:

How can we design tools and learning environments for musical tinkering?

How does musical tinkering unfold as a learning process?

As I explore these questions, I will use the ideas of loops and maps both to explain my design decisions, and to analyze the learning processes I observe. In the chapters that follow, I first present the design of MelodyMorph, an iPad app for making musical instruments and compositions on the screen. Then I review case studies of middle school students tinkering with the app, with one in-depth story of a single student and a few shorter vignettes. Next I present the design of MaKey MaKey, an invention kit that can be used to create physical-digital musical instruments. I give a detailed categorization of projects people have done with MaKey MaKey "in the wild," and review case studies of my own musical tinkering workshops using the kit, and of other educators using it. Finally, I conclude with reflections on the future of musical tinkering.

Chapter 2:
Designing MelodyMorph

In this chapter I present the design of MelodyMorph, an iPad app that supports musical tinkering. I describe the inspirations for the project, introduce the app's functionality in detail, and discuss the concepts that guided its design. Then I give examples of MelodyMorph activities, for both classroom and informal contexts. I discuss the novel affordances of MelodyMorph, review some related work, and conclude with ideas for the future.

Inspiration

The initial inspiration for the design of MelodyMorph came from physical manipulatives called Montessori bells, and the music creation software tools Impromptu and Hyperscore.

Montessori Bells

Jeanne Bamberger's work with Montessori bells (Bamberger, 1991) is the main source of inspiration for MelodyMorph. The Montessori bells are a set of metal bells on wooden stands. They all look alike, but play different notes when tapped with a mallet. Bamberger created musical tasks for children to explore using the bells, such as constructing the tune "Twinkle, Twinkle, Little Star." As children work through the tasks, the confusions and puzzles that arise provide a window into their musical thinking. In my exploration with the bells, I wondered what it would be like, instead of having just a dozen of them, to have hundreds of bells. What new puzzles would appear? What new creative possibilities would I find? If the bells could be arranged on a screen, perhaps I could use them to make a new hybrid that could be both a musical composition tool and a reconfigurable instrument.

Figure 1: Montessori bells are physical bells that look alike, each with a different pitch (photo by David Veksler).

Impromptu

Impromptu is Jeanne Bamberger's environment for experimenting with "tuneblocks," which are graphical blocks representing short melodic fragments, typically a few notes long and

forming only part of a musical phrase. By putting the tuneblocks into a sequence on the screen, learners can reconstruct existing tunes, or experiment with composing their own. As with Montessori bells, the tuneblocks provide a way to explore musical thinking. Because Impromptu starts at a higher level of abstraction (melodic fragments instead of individual notes), and uses several different symbolic representations, it can be used to build up more complex musical structures. The focus of Impromptu, and its accompanying music theory text, *Developing Musical Intuitions* (Bamberger, 2000), is more on understanding musical ideas than on musical expression. It focuses more on musical inquiry and experimentation than on performing or composing.
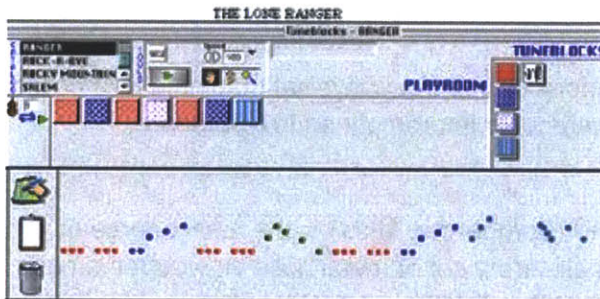
Figure 2: Impromptu software. The tuneblocks are the horizontal row of colored squares, each representing a melodic fragment. Melodic contour notation is shown below: a colored dot for each note, with pitch on the y axis and time on the x axis. Each red tuneblock above represents one group of six red notes, below.

Hyperscore

Hyperscore (Farbood et al, 2004) is a powerful software environment for composing by novice users. The user creates a musical motive by placing individual notes in a window with a grid representing pitch and time. They then draw a line in a sketch window which represents the motive repeating over time, shifted up or down in pitch according to the height of the line. Lines for several motives can be drawn using different colors, and the generated pitches are automatically adjusted so that the concurrent motives harmonize with each other. A "harmony line" also allows the user to shape the chord progression used to harmonize the motives. This tool, like Impromptu, also affords detailed exploration of musical ideas, but its emphasis is more on creative expression. It provides fewer levels of symbolic representation, and hides some musical details by automating them. But it also provides a richer palette of instrument sounds and enables the creation of more complex polyphonic compositions.
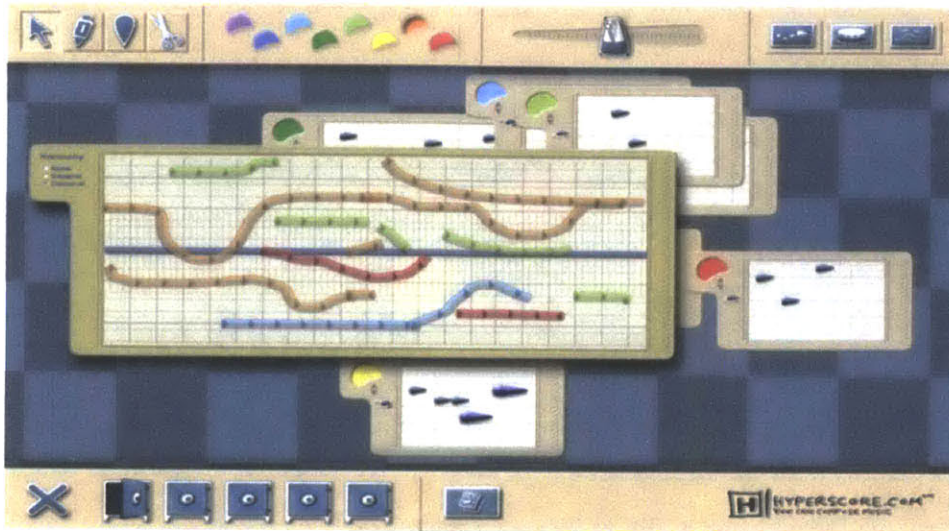
Figure 3: Hyperscore software. In each small "melody window" the droplets represent individual notes. The colored lines in the larger "sketch window" correspond by color to the melody windows, and represent a composition that shifts them in pitch using the height of the line.

Functionality

MelodyMorph is an iPad app for constructing playable musical instruments and compositions on the screen. It is implemented using OpenFrameworks, an open source C++ toolkit for creative coding. Inspired by Montessori bells, MelodyMorph provides a set of objects on the screen that you can drag around and tap to play different pitches. Because of the app's focus on creation, MelodyMorph's bells, unlike the physical bells, are labeled with color and texture to indicate their pitch (with an optional text label showing the pitch class, such as F#, as well). Numerous other features provide tools for constructing, editing, and performing in ways that extend beyond what would be feasible even with a large number of physical bells.

Bells

At the top of the MelodyMorph screen, a palette contains a row of objects representing different pitches, which I refer to as "bells" (see top of figure 4). Colors and textures indicate the pitch of each bell: each octave is a rainbow of hues, with C major triad notes having a solid color, the other diatonic C notes having a single stripe, and non-diatonic notes (corresponding to the black keys on a piano) having vertical stripes (figure 5a). Optional settings enable showing and hiding the non-diatonic notes and the pitch class labels. The palette can be shifted up or down an octave, with the lower octave in a darker version of the same colors, and the upper octave in lighter colors (see figure 5a). The palette has tabs to switch between several instrument timbres, which are represented by the shapes of the bells: vibraphone bells are triangles, piano bells are round, pluck bells are starbursts, etc. (see figure 5b).

Figure 4: MelodyMorph. The app screen, showing the palette of bells at the top, with the piano instrument selected. Along the left side of the screen are the mode buttons for (from the top): full-screen, drawing, erasing, selecting, path-making, muting, and strumming. Along the right side are buttons for (from the top): stopping all recordings and loops, showing the help screens, opening the options panel, and toggling record mode.
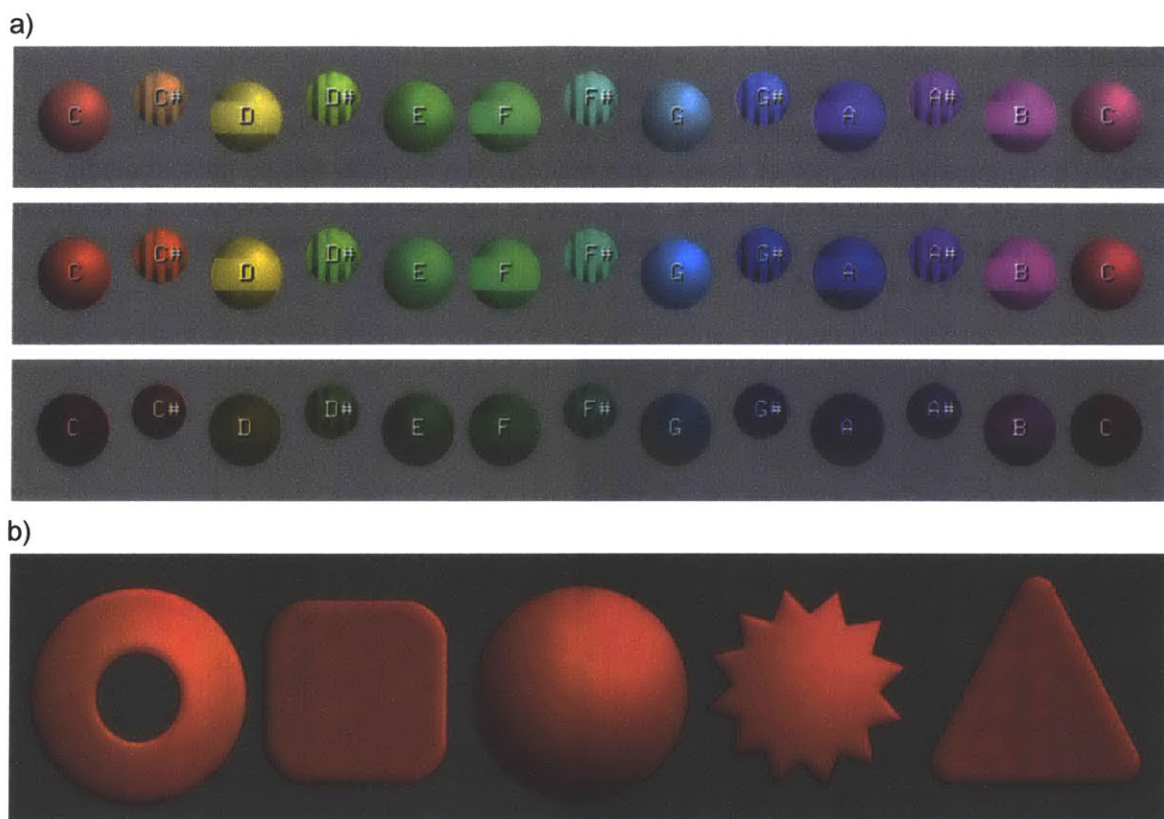
Figure 5: Bell colors and shapes. a) The palette of piano bells, labeled by pitch. The three octaves available are shown: the upper octave are in brighter colors and the lower octave in darker colors. b) A bell for the note C in each of the five instruments available. From left: bass, electric piano, piano, pluck, and vibraphone.

Canvas

Bells can be dragged from this palette onto a canvas that is zoomable and unlimited in size. Bells play their note when tapped on the canvas or in the palette. Once the bells are positioned on the canvas, they can be played by tapping them individually, or together to play a chord. Unlike typical music software tools, MelodyMorph does not assume or impose any particular mapping between space and other musical parameters. Typically, for example, a sequencer will use the horizontal axis on the screen to represent time, and the vertical axis to represent pitch, or an instrument controller will map pitch to one axis and a sound synthesis parameter to the other. Instead, I was inspired by the way the children observed by Bamberger used spatial arrangements of Montessori bells to represent a variety of musical structures, including melodic sequences, pitch relationships, and phrase boundaries. The canvas in MelodyMorph allows learners to use space as space: they can arrange their creation in a way that reflects their own understanding of musical ideas, and create their own pathways to move through them.

Recording

A record button allows the user to play a melody on the bells, which then becomes a new object, a recorder bell. The recorder bell behaves much like the other bells, except that tapping it triggers the playback of its recorded melody. The recorder bell contains melodic contour notation of the recorded melody: each note is represented as a colored dot, with pitch on the y axis and time on the x axis. The notation animates as the recorder bell plays, each note appearing as your hear it.
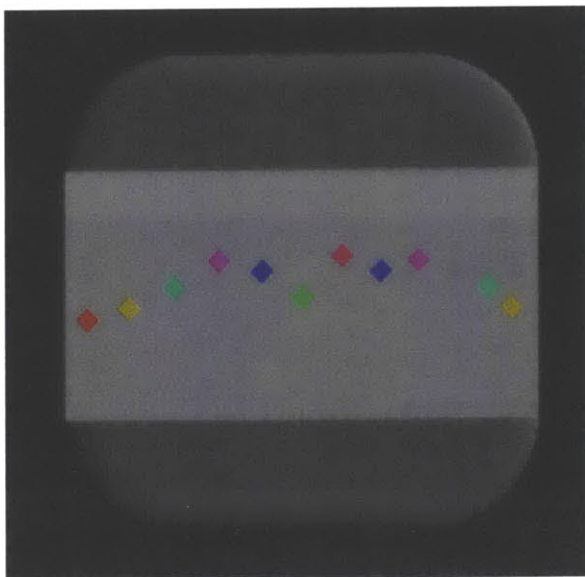
Figure 6: A Recorder Bell containing a short melody. Pitch is on the vertical axis and time is on the horizontal axis. Each diamond represents a note. Their colors match the bell color scheme. This recording corresponds to a performance of the ring melody in figure 4.

Editing

MelodyMorph's editing modes include drawing and erasing, strumming, path-making, and selection. The drawing and erasing modes allow the user to annotate their creation with freehand lines on the canvas. These can include, for example, words, symbols, decorations, or narrative elements (see examples in figure 8). For example, I have seen people use it to draw characters in a story, build elaborate musical landscapes, and make text and arrows that show how to move through a melody. A "strum" mode allows the user to slide their finger over the bells to trigger them (otherwise, this drag gesture moves the underlying canvas). The "path-making" mode records this sliding motion on the canvas, drawing it as a line with a button at the beginning (see figure 7). Tapping this path-button causes the path to play back. A triangular path-head follows the path's line over the canvas, repeating the same speed and timing as in the recorded gesture. The path-head triggers any bell that it touches, so that bells can be moved around after the path has been made, changing the resulting melody. Path-heads can also trigger path-buttons, allowing complex structures to be created, including

branching and infinite loops. Path-heads can trigger recorder bells as well. Finally, the selection tool allows the user to drag groups of bells around on the canvas, copy and paste them, and shift their pitches up and down as a group.
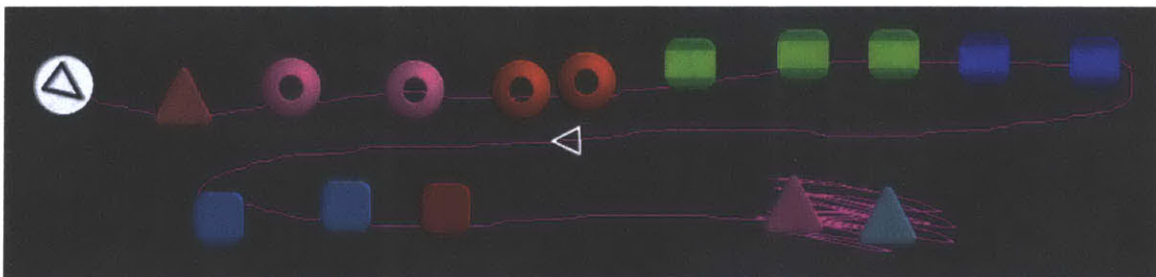
Figure 7: A path created with the path player. The user started at top left, playing a melody by sliding a finger across the row of bells to the right, then slid back across to the left, then to the right over the row of three bells, and finally wiggled rapidly back and forth over two vibraphone bells to make a bell-ringing sound effect. Touching the path player bell at top left triggers the path head (the white open triangle at the center) to replay the path, with the same speed and timing as the original gesture, triggering the bells as it touches them.

Examples

MelodyMorph can save and load users' creations ("morphs"), and load example morphs. The app comes loaded with a set of examples showing a variety of ways to use it. These include (see figure 8):

- A **cyclic melody**, arranged in the shape of a ring
- A **kalimba-like instrument** with pitches arranged in a V shape, ascending in diatonic thirds to mimic a traditional thumb-piano
- A deconstructed **pop song**, with a drawing of the band's logo, and melodic and harmonic material from the song positioned so that it can be played in different ways
- A **jazz standard**, with the melodic notes lined up with accompaniment chords and bass lines so you can play them all at once. A second representation of the song is divided into chord changes, with a separate annotated scale for each one, making it easier to improvise over the changes.
- A **cartoon dialog** between two characters, with melodies in speech balloons
- A section of a Chopin **piano piece**, laid out so that it can be performed by quickly sliding a finger over cascading rows of notes
- A **network of paths** that trigger each other in branching and looping patterns, triggering a complex melodic texture
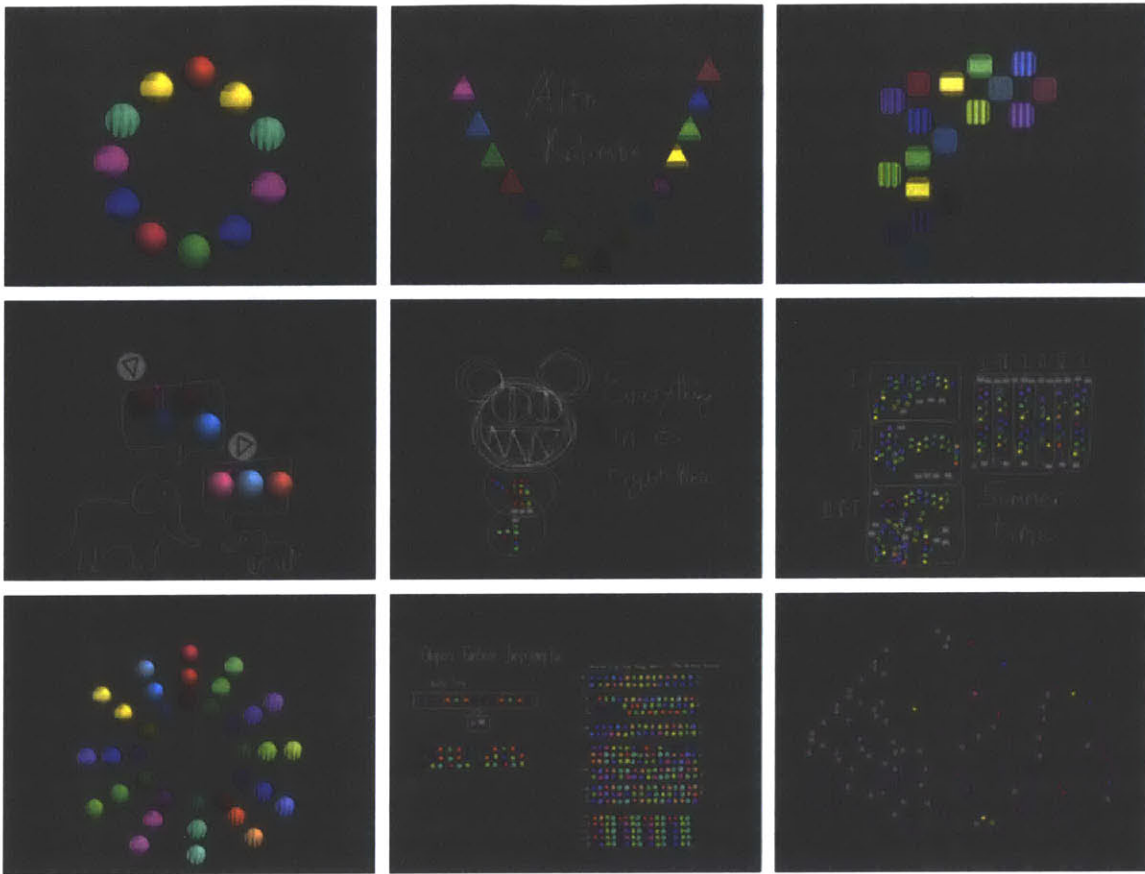
Figure 8: Examples. From top left: a ring melody; a thumb-piano kalimba-like instrument; a minor scale instrument; a musical cartoon; a pop song; a jazz standard; the circle of fifths; a Chopin excerpt; a network of paths.

Design Concepts

After designing MelodyMorph, I can see in retrospect that several concepts guided my design. These are all within the context of designing for constructionism, which inspired me to focus on making a creative tool for musical making. I was especially influenced by Resnick's (2005) concept of designing for low floor, high ceiling, and wide walls. The app aims to support low floors by making it easy to get started, high ceilings by making it possible to make complex creations, and wide walls by supporting a variety of themes and interests.

In addition to these ideas, I can identify several others that implicitly guided more specific aspects of my design. I will describe these using the concepts of tinkering loops and tinkering maps. Three principles of designing for tinkering loops are immediate feedback, scaffolded complexity, and authentic context. For tinkering maps, they are flexible space, expressive gesture, and linked symbols.

Immediate Feedback

Tinkering loops involve quick, repeated experiments. Ideally these form a continuous flow of interaction, creating a conversation with musical material. As Resnick and I point out in our chapter on "Designing for Tinkerability" (2013), a tinkerable design lets the user both see the results of their experiments immediately, and see into the internal processes of the system as it works. MelodyMorph allows the user to see immediate results mainly through its primary interaction: tapping a bell in order to play its sound immediately. In a typical music composing interface, the user moves symbols around on the screen, then presses play to listen to the results, then moves the symbols again. Compared to writing a score by hand, and bringing it to an orchestra the next day, this is immediate feedback. But MelodyMorph shortens the feedback loop further, combining arranging and playing into a single set of gestures that produce sound. Two other features of the app also allow the user to see its processes working. The recorder bell, as it plays back a recorded melody, reveals the notes one by one in the melodic contour notation that appears on it. The path player shows a triangular path head moving along the path and triggering bells as it recreates the users movements. Being able to watch these processes makes it easier to understand and modify them.

Scaffolded Complexity

There is a large space for learners to navigate between the app's low floor, for easy entry, and its high ceiling, for reaching more sophistication. The growth between these is the world of tinkering in context-loops, in which progress is non-linear, and feedback comes from many sources, especially other people. Some features of MelodyMorph are designed to help people gradually develop as they use the app. A typical approach to "scaffolding" in educational technology is to create a highly constrained interface at the beginning, with explicit and specific instructions, and little opportunity to deviate from a fixed path. This scaffolding is then gradually removed as the user progresses. Instead, in MelodyMorph, the app maintains a consistent appearance and feature set. The user can make of use of different resources that act as "scaffolding on demand," such as a sequence of help screens, and a set of examples ordered by their complexity. These features enable users to find their own way within the app, learning new features as they choose. More importantly, a set of activities designed for a collaborative learning environment provides a context for learning through tinkering loops.

Authentic Context

For a tinkering process to enter the slow, powerful outer-loop, it has to be connected to the authentic inner and outer worlds of the tinkerer. This means that the materials of the tinkering activity must be personally meaningful, and connected to the mental world of the tinkerer. MelodyMorph allows users to pursue musical ideas based on their own understanding, and create projects by following their own interests. The process must also involve authentic interactions with the surrounding environment, especially the social world. MelodyMorph can be used for creative collaboration, group improvisation, and musical performances for peers.

Flexible Space

In order for a tool to support the creation of tinkering maps, it must support the externalization of ideas and thought processes. This requires a flexible use of space. As I mentioned in the description of the canvas features of MelodyMorph, the app does not impose any specific relationship between spatial parameters (such as position on the x and y axes) and musical parameters (such as time, pitch, or loudness). These fixed relationships are typical of computer music interfaces. Instead, MelodyMorph enables the user to make flexible and creative use of space and spatial relationships. This idea comes in part from my observations of Montessori bells: their sound of course is not affected by their location on the table. Instead, children can move them around and use their spatial relationships creatively, for example to support different gestural paths across them as they play them, or to represent musical relationships such as pitch intervals and phrase boundaries. MelodyMorph keeps the use of space flexible so that people can position the bells on the canvas in these ways. It also supports the more complex uses of space afforded by having a limitless canvas and number of bells, such as the creation of musical scenery, musical diagrams (such as a playable circle of fifths or an array of scales), or other arrangements.

Expressive Gesture

In order to support the creation of tinkering maps that reflect meaningful ideas, a tool must support intuitive and expressive creative processes. MelodyMorph is designed to support expressive performance of music, with an immediate causal relationship between physical gestures and sounds. Many computer music interfaces support only more abstract gestures, such as positioning notes on a graph of time and pitch. The computer interface is used primarily as an editing interface, with expressive gestures possible only on more traditional instruments connected to the computer. In order to incorporate expressive aspects of musical instrument performance into my app, I included a few ways in which expressive gestures translate into sound. Tapping a bell triggers its sound immediately, so playing a melody often involves a sequence of tap gestures with expressive movement and timing. The "strum" mode allows the user to slide their finger across the screen to trigger bells, a more continuous gesture. The recording mode captures and replays the specific timing of tapping gestures, and the "path" mode similarly records and replays these sliding gestures, preserving their speed and timing.

Linked Symbols

To create tinkering maps a tool must have the power to link symbols and actions. This means a user can use the tool to create a symbolic representation within the world of the tool that can also have an effect on the world outside of it. In the case of MelodyMorph, this link is between the arrangements of bells, recorder bells and paths on the screen, and their behavior, which is the generation of musical sounds.

Classroom Activities

We cannot isolate the design of the tool from the way it is used in a learning environment. Here I present several activities that I have created to structure the MelodyMorph learning experience. The following are activities that we tried with 5th grade students in the course of the study I describe in the next chapter.

Treasure Hunt

This initial activity provides a way for children to explore the app and discover its features without too much explicit guidance. We give a brief introduction and a demo of the app on a projector screen, and then post several questions on the board in front of class for the students to answer, such as "how many different sounds can you get it to make?" and "what do the colors and shapes mean?" We hand out the iPads, and the students work in small groups, often calling out their discoveries as they make them, and showing them to each other across groups. After a few minutes, we have a full group discussion with the class in which they share their discoveries, and we show them a few things that they may have missed.

Musical Puzzles

We created musical puzzles for the students to solve, drawing on knowledge of musical compositions that the students had recently been studying in the same music class. For one puzzle, we asked the students to recreate a short melody which they had learned to play on the piano, the bass line from the song "Stand By Me." In that case, they assembled the melody using individual bells in the app. In another puzzle, we provided a set of recorder bells on the canvas, each containing a fragment of this same melody, but in a scrambled order. This puzzle is inspired by one of the tasks used in the Impromptu software, reconstructing tunes using tuneblocks, which similarly contain melodic fragments. The students had to rearrange the recorder bells on the canvas in the order of the original melody, and play them directly, or record a path that triggers them each at the correct time.
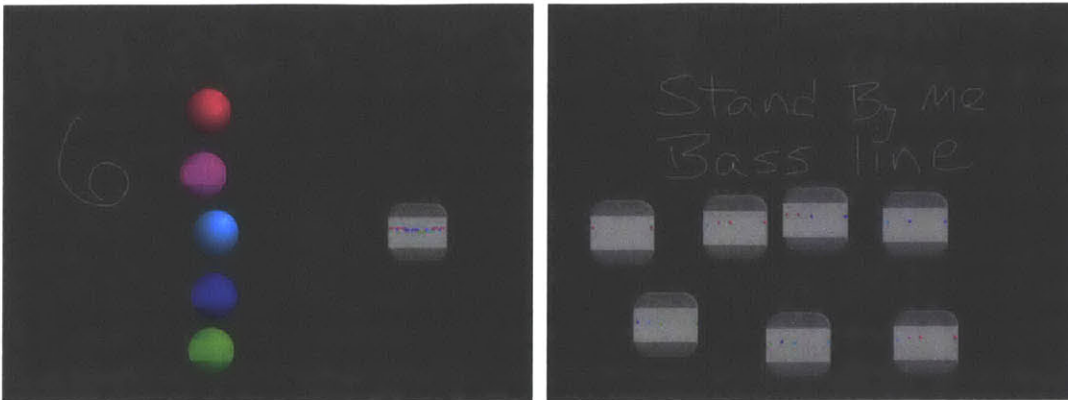
Figure 9: "Standy By Me" bass line puzzles. Left: All the individual bells required to play the bass line, and a recorder bell containing the correct bass line; Right: Recorder bells containing fragments of the bass line to be put in order.

Musical Cartoons

In this more open-ended activity, we asked the students to create musical cartoons. In our examples we showed each character with a speech balloon. Instead of text, the speech balloons contained melodies. For example, a mother elephant speaks with a low pitched piano melody, and the baby elephant replies with a similar melody in a higher octave (see figure 8). This technique could be used to tell a story in several scenes, as in a comic strip. The students invented their own variations on this technique, combining drawings and melodies in other ways. These included making the colorful bells part of the imagery (for example, using the round piano bells as eyes, or a triangular vibraphone bell as a hat), and using the bells to make sound effects instead of melodies (such as a heartbeat, or a helicopter sound).
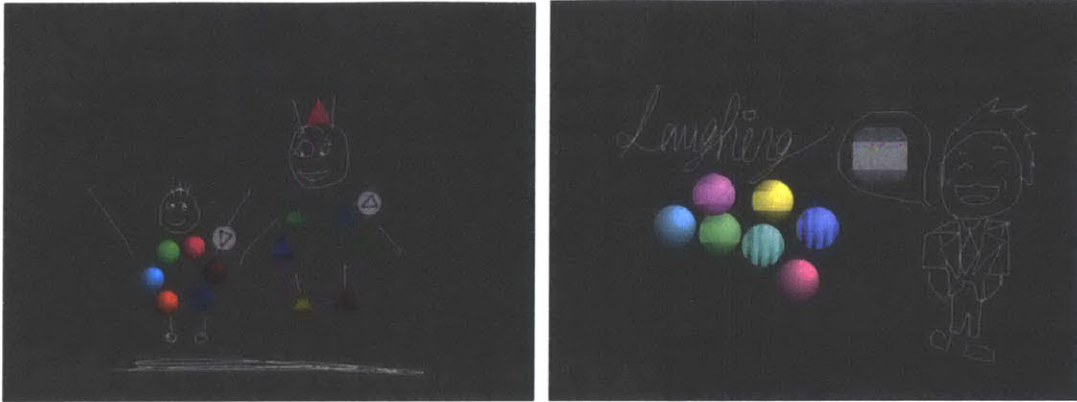
Figure 10: Musical cartoons created by students. Left: two characters with bodies created partly from bells. Each person is made from a melody that can be played by a path player. Right: A laughing character, with a melody in the recorder bell inside the speech balloon, alongside the upper-octave piano bells used to create it.

Scale Palettes

In one version of the musical cartoons activity, we introduced different musical scales for the students to experiment with. We created a template project, with four scales provided as rows of piano notes: major, minor, harmonic minor, and whole tone. We showed how these could be used to make melodies: instead of pulling bells from the palette, they could remove bells from the scales (the scales were actually made up of bells stacked on top of each other so as to simulate the functionality of the palette). We did not explain the meanings of the names of the scales or their typical uses, instead allowing the students to find their own ways to use them.
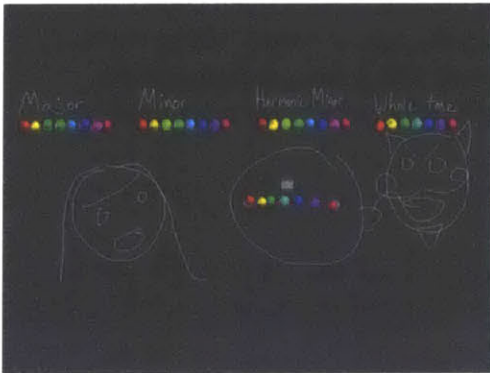
Figure 11: Scale palettes with an example. In a template we provided, there were four scale palettes to try out: major, minor, harmonic minor and whole tone. In this student's cartoon, they have used the whole tone scale to make an evil cat's scary melody.

Musical Stories

In the musical cartoons activity, we found that many students spent more time using the drawing tools than they did using the musical tools. To encourage more musical exploration, we created another activity in which they told stories with music, but this time without drawings. We asked the students to describe a sequence of events in words, and make a melody for each part. For example, one group made melodies for different parts of their day: "morning," "school," "sleep," and "eating."

Figure 12: Musical story. The students created melodies for different parts of their day.

Open-Ended Projects

After completing the other activities, we gave the students two class periods in which to work on projects of their own choosing. We started with a review of the previous activities, and a brief brainstorm about what kinds of projects they might want to make. We concluded with presentations of the projects from each group for the whole class. The resulting projects included a musical story about going to math class (and happily leaving math class), an interpretation of the theme music from a classic fighting video game called "Mortal Kombat," elaborate musical cartoons such as a helicopter adventure, and a reconstruction of the multilayered music from a mobile game called "My Singing Monsters," with accompaniment on a bongo drum.

Informal Activities

More informally, I've experimented with several other types of activities including musical dialogs, instruments and systems for improvisation, group improvisation, tune reconstruction, and music theory experiments.

Musical Dialogs

With a group of graduate students, I tried a variation on the musical stories activity in which the students created stories based on spoken dialog. They made up brief back and forth exchanges with specific wordings, and constructed melodies in which each spoken syllable corresponds to a single note. Structuring the melodies in this way can help people even

without musical training to create melodies in a meaningful way, by drawing on their knowledge of the timing and melody of speech, and by comparing the sound of speech to their melody as they compose.

Improvisation Instruments

Personally I've spent quite a bit of time making improvised music with MelodyMorph, in private and also in performance for an audience. One strategy I've used is building a specialized instrument containing its own musical landscape, such as a cluster of notes from a minor scale that is not in a typical pitch ordering. Exploring this landscape can lead to new musical ideas, by allowing me to release myself from my prior knowledge of a scale and its spatial patterns. Instead I can stumble intuitively into new melodic possibilities, and listen for them more openly, without attachment to my expectations.

Improvisation Systems

I've also built up systems in MelodyMorph for improvising over a particular accompaniment pattern playing in a loop. A simple version uses a loop with no chord changes, allowing me to improvise over it with a single instrument designed for that purpose. I have observed other people (generally with some musical background) on several occasions developing improvisation systems of this type with MelodyMorph. I've also begun to experiment with a more complex type of system with a loop that moves between chord changes, requiring the improviser to move between separate scales on different parts of the canvas. One such system that I've created plays accompaniment for the chord changes to the jazz standard "Summertime," with a separate scale for each chord. The scales in that case are laid out with the chord tones slightly offset from the rest of the notes, and the root notes are circled.
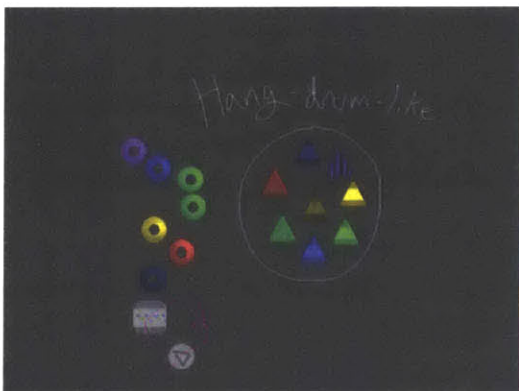
Figure 13: Instrument with accompaniment loop. In the circle on the right, an instrument made up of a cluster of notes inspired by the pitch arrangement of a Hang drum. On the the left are bass notes for accompanying it, a recorder bell containing a bass line, and a path that triggers it in an infinite loop.
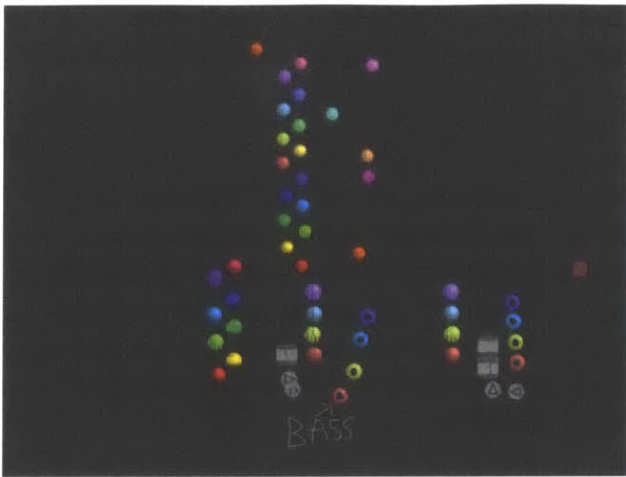
Figure 14: Jazz instrument with accompaniment. This project, created by a 15 year old with jazz piano experience, has a looping accompaniment pattern at the bottom, including a bass line and piano chords. The columns of notes at the top make up an instrument for improvising over this loop with pluck sounds. The main column of notes is a minor scale, with the chord tones offset to the right of the other notes in the scale. Further to the right are chromatic notes in between the scale notes, to be played with the right hand.
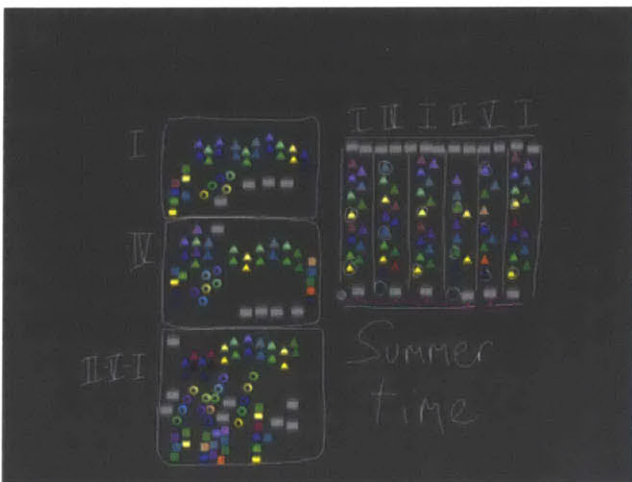
Figure 15: Summertime arrangement. On the left side, the boxes contain the melody to this jazz standard, in octaves, along with recorded accompaniment chords and bass line. On the right side, a path triggers segments of the bass line for each chord change, with a scale in a column above each for improvising over it.

Group Improvisation

I've experimented a bit with improvising with two or three people, each on their own iPad running MelodyMorph. One approach is to choose a simple constraint at the beginning, such

as each person selecting one of the instrument sounds. Initially the improvisation tends to unfold through conversational turn taking, watching each other's screens. This conversation can evolve into a repeating pattern in unison or in harmony, or one player might shift to accompanying pattern while the other improvises. Other times the group may play together in a more free-form way with only a loose coupling between the players in rhythm and harmony.
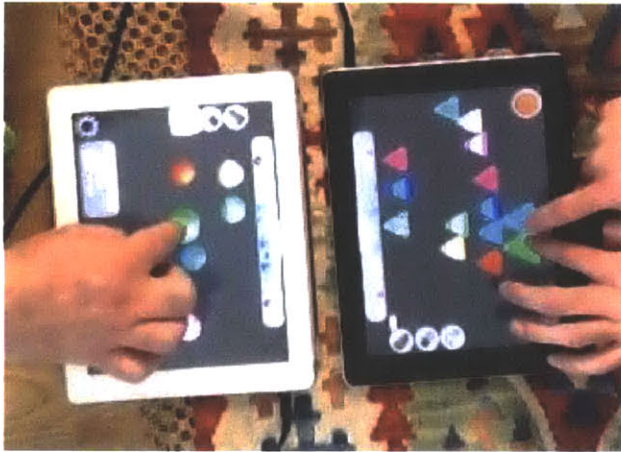
Figure 16: Duet improvisation. MelodyMorph on two iPads, used by me and a musically experienced friend to play an improvised duet. We use different instrument sounds, but have similar arrangements of bells on our screens. At the moment of this freeze frame, we are playing the green note (an E) simultaneously.

Tune Reconstruction

In several cases people using the app have spontaneously decided to recreate a tune that they know. In some cases these are children reconstructing music from their favorite video games (examples I've observed have included the themes from "Super Mario Brothers," "Angry Birds," "Mortal Kombat," and "My Singing Monsters"). In some cases they have had experience learning this melody on another musical instrument. I have also observed people reconstructing melodies from folk tunes or from popular music. In my own experience, inventing a spatial structure to represent the harmony or song form often leads to a new understanding of the song by revealing its structure. For example, my reconstruction of the Radiohead song "Everything in its Right Place" helped me to see that the two of the sections use the same chords but in a different order, and a third section adds just one other chord to the first set (see figure 8).

Music Theory Experiments

I have not experimented in a formal learning environment with using MelodyMorph to explicitly introduce music theory concepts (except perhaps in the activity including scale templates). But I have informally experimented with several concepts in this way. I have explored motivic variation by making a melody, then copying it and transforming it in different ways, such as transposing it, reordering the notes, and changing individual notes. I have also constructed

spatial pitch patterns representing music theory diagrams. These have included a circle of fifths, a sequence of major triads in diatonic thirds (mimicking the arrangement in a popular music app called "SoundPrism"), and chromatic rows of chords, each with multiple voicings. Each of these can become a new kind of instrument, or a music composition tool to help find new melodic patterns and harmonic possibilities.
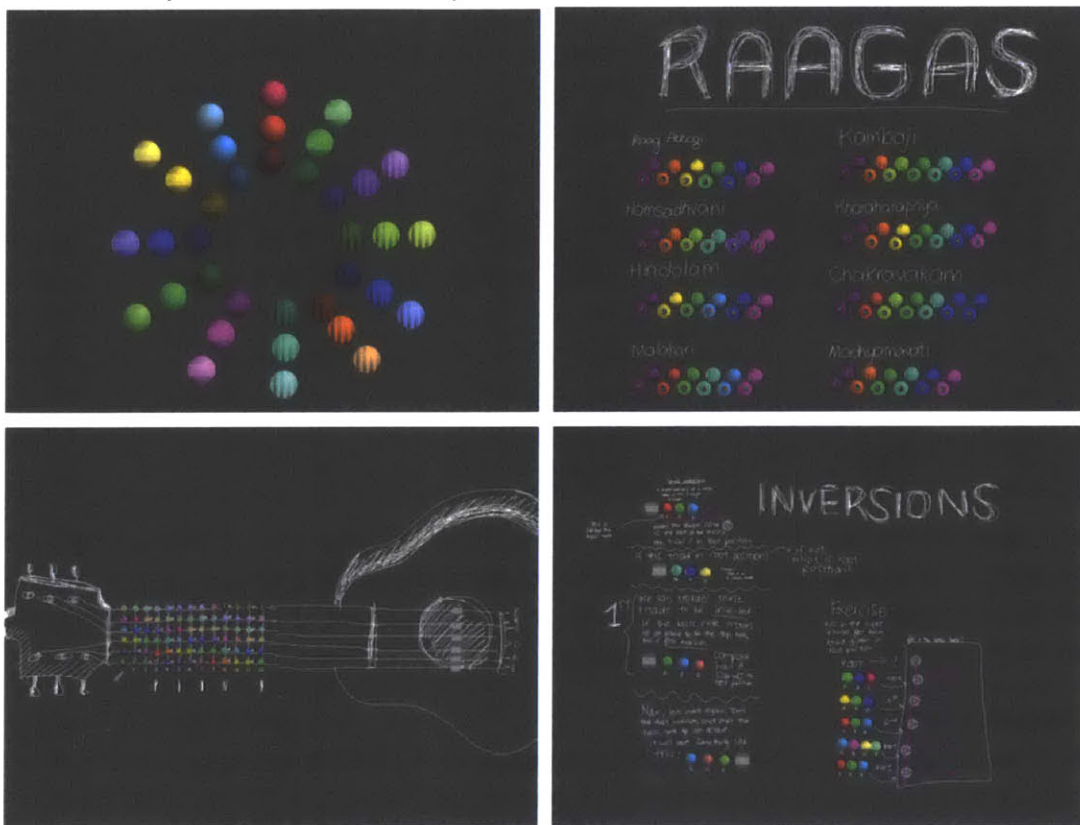
Figure 17: Music theory experiment examples. Top left: the circle of fifths. Top right: versions of several Ragas, scales used in Indian music. Bottom right: an interactive tutorial on chord inversions. Bottom left: a guitar with bells for each note on the fretboard.

Novel Affordances

In the course of experimenting with MelodyMorph on my own, with friends, and in formal and informal learning environments, I've discovered that it has some novel affordances for music making, beyond the design concepts described above.

Multi-Modal Musical Storytelling

MelodyMorph supports storytelling that combines drawings, such as a cartoons and scenery, with musical material such as themes, background music, and sound effects. While other tools exist for multimodal digital storytelling, these generally focus on images and video, and are not primarily music composition tools.

Time-Structured Instruments

MelodyMorph allows people to create their own instruments with flexibility not only in spatial structure, but also in temporal structure. A MelodyMorph canvas can contain separate instruments in different locations for improvising over different accompaniment patterns or chord changes. It can also dynamically illustrate this structure in time, through use of the path player feature. One way I've experimented with doing this is by creating recorder bells containing short segments of accompaniment, and using a path player to trigger them in sequence. By watching them to see which part is currently playing, the improviser can follow along and play on the corresponding region of the instrument.

Visual Communication for Improvisers

Improvising in a group with multiple iPads revealed an unexpected benefit of the visual language of color for pitch in MelodyMorph. Each person can tell which notes the other is playing by looking at their screen, and then match the colors to the bells on their own canvas. On more traditional instruments, improvisers typically work out the melody another person is playing by ear, which requires skill in relative pitch perception. Alternately, with sufficient knowledge of another instrument, one musician can watch the other's fingers to work out what they are playing. With MelodyMorph, it is possible to match the pitches from another person's melody just by looking at what is on their canvas. Unlike the fingerings on a traditional instruments, these notes persist in place, leaving more time to see them. This affordance could help to make group improvisation more accessible to beginners who have not yet developed skill in playing by ear.

Playable Diagrams

Tools for illustrating music theory ideas and for diagramming analyses of musical pieces generally result in static visual artifacts. MelodyMorph makes it possible to create an analysis of a piece that is playable, supporting more intuitive exploration of musical ideas. In this way, educators could use MelodyMorph to create interactive music theory learning materials. Learners can also use MelodyMorph as a tool for musical inquiry, building up their own diagrams to test musical hypotheses and demonstrate their ideas.

Design Challenges

Several problems and questions have arisen in the development of MelodyMorph that are still unresolved. They are in the areas of user interface design, expressive interactions, visual representation, audio design, and support.

Preventing mode errors

One problem with MelodyMorph's interface is a type of error people make called a "mode error" (Norman, 1983). Mode errors occur when one action, such as a pressing a button, can have more than one result depending on the current state, or mode, of the software. A familiar example in a text editor occurs with font-style modes such as bold or italic. If the "bold" button is toggled on, any text you type will be bold. A mode error can occur when you toggle it on,

type some text you intend to be bold, then start a new line and type more text. This new line will be bold even though you intended it not to be, having forgotten to press the bold button again to toggle it off. In MelodyMorph, there are several modes that cause tap and drag gestures on the screen to be interpreted in different ways. By default, when no mode is selected, tapping a bell plays it, dragging a bell moves it, and dragging the background moves the canvas. Other modes allow you to do things like tap bells to delete them, or drag on the background to draw lines.

I hoped to prevent mode errors in MelodyMorph by using a type of mode called a "quasi-mode," in which the mode is not toggled on and off by a button, but instead the user has to hold down the button the entire time they wish to use the mode. The most common example in current operating systems is the shift key, which you hold down to type capital letters. The idea is that quasi-modes prevent mode errors by preventing you from accidentally leaving the mode on after you are done with it, since you will naturally remove the finger from the mode button. Quasi-modes are inherently "multi-touch" interactions, and generally require two hands. This makes them a natural fit for multi-touch devices such as the iPad. They are nonetheless not currently a standard part of the design vocabulary, so even fluent users of these devices do not expect buttons to work this way. People can quickly learn the idea, especially when another person demonstrates it, but it is not easy to make a clear affordance for it that users can perceive within the interface.

Most people do not discover the quasimode functionality in MelodyMorph on their own. Many people also learn it but then revert, when distracted, to their earlier understanding of mode buttons as toggles, tapping one of them and then tapping the canvas. For these reasons, I will likely remove the quasi-modes from the app, and find another solution. This might involve changing the buttons to work as toggles. In that case, I will hopefully find an alternate way to communicate the mode state to the user, using sounds (as in e.g. Monk, 1986) or graphical cues such as a change to the color or texture of the canvas. Also, as Norman (1986) suggests, it is best to make editing actions reversible, through a mechanism like an "undo" system that reduces the damage caused by mode errors.

Preventing trouble with zooming

Another user interface problem with MelodyMorph results from zooming in and out on the canvas. At the default zoom level, the bells are a reasonable size target for quickly tapping them to play melodies. Zooming out to the maximum zoom level allows you to see a much larger area, and construct more complex arrangements. But the bells become so much smaller on the screen as a result that people frequently miss them, and fail to trigger a note. One solution would be to decrease the maximum level to which you can zoom out.

Limiting the maximum zoom-out level would also cause another problem. It would become impossible to see an overview of the canvas. To help with this problem, I could provide an alternate view, such as a tiny map showing the entire bounds of canvas which has been used so far. In a very early version of MelodyMorph, I had such a "mini-map," because I had no

zoom functionality yet. I later removed the mini-map, because it did not seem necessary, when I was not yet able to build very large or complex creations. Now that its usefulness seems clearer, perhaps I can add something like it back in again. Another problem it would help to prevent is the situation in which a user positions some bells, then moves to a distant location on the canvas, and is unable to find their earlier work.

Using motion-sensing to enhance expressivity

The expressive quality of MelodyMorph is an important aspect of the design. I am still hoping to develop two features for a future version that make use of sensing to improve expressivity: tap force sensing, and vibrato. Generally, when you strike an acoustic instrument, like a piano key or a string, a harder hit causes it to make a louder sound. I have experimented with simulating this effect on the iPad by using the signal from the built-in accelerometer, measuring the spike in the z axis (perpendicular to the screen), which represents the jolting movement of the device when you tap it. Several problems with this approach have made it unreliable. These include the difference in tap force between people, the difference in the accelerometer signal depending on whether the iPad is held in the air or is lying flat on a surface, and the delay in the accelerometer signal after the screen has registered a touch.

In an earlier version of MelodyMorph, I included a feature that allowed the user to create a vibrato effect by holding down a bell after it has been tapped and wiggling the iPad back and forth. I measured the acceleration in the plane of the screen, and proportionately modified the pitch of the sound as it played. The effect was very immediate and expressive, though many details needed improvement. These included scaling the pitch shift amount correctly for each pitch range, and ensuring smooth pitch changes without glitches. In a later version, I rebuilt the sound engine, and had to remove this vibrato effect, but it should be possible to add it back in again. One drawback of this design is that it requires the iPad to be held up with one hand and played with the other. Often, MelodyMorph is actually used with two hands while the iPad rests on a table. It may be possible to simply measure small finger movements on a bell after it has been tapped, and use those to trigger vibrato instead.

Choosing colors, textures, and shapes

My choice of the mappings between visual and sonic properties in MelodyMorph are fairly arbitrary. I chose to map hue to pitch class, brightness to octave, and shape to timbre. These relationships made intuitive sense to me, but I did not have a theoretical basis for them. There is no single standard relationship between color and pitch. Historically, composers have described their own systems, each unique (e.g. Berman, 1999). I chose to map a rainbow of hues, starting with red, to an octave of pitches, starting with C. I chose these two as the obvious starting points, but oddly enough, they create a conflict for me.

I personally have a version of synesthesia, the crossing of sensory modalities in which a sensation in one can trigger a sensation in another. This synesthetic experience is probably the original case for my fascination with the relationship between music and visual imagery. In my case, letters and numbers are each associated in my mind's eye with a specific color.

When I think of the letter C, I imagine it in yellow. The fact that the pitch C is instead red in MelodyMorph fortunately does not create any cognitive dissonance. I also use stripe patterns that differentiate triad, diatonic, and non-diatonic notes. I added the stripes out of necessity, because it seemed difficult to distinguish more than eight or so different color categories. The stripes make it easy to see the difference between two bells which would otherwise be two very similar shades of green, for example. One potential problem with this color mapping is that people with color-blindness are left with less information. This problem is mitigated by the availability of pitch labels for each bell.

It would be interesting in future versions to experiment with allowing users to specify their own mappings between pitch and color, or between timbre and shape. It would also be interesting to have a setting that removes the visual differences between the bells entirely. This would make them more faithfully imitate their original inspiration, the Montessori bells, and support similar explorations of the pitch property isolated from other characteristics.

Synthesizing instrument sounds

I have redesigned the audio synthesis engine for MelodyMorph twice. The app must be able to play a large number of voices simultaneously, in several different instrument sounds. One challenge this creates is in mixing the audio in order to prevent clipping distortion when they become too loud. Another challenge is smoothly transitioning between a fixed number of voices using a "voice-stealing" algorithm, as large numbers of notes play at once.

Synthesizing several distinct instrument timbres also presents a challenge. I chose to use a simple time-stretching algorithm to control the pitch of a sampled instrument sound. This method works better for some instruments, such as the vibraphone, than others, such as the piano. In some cases, I use a multi-sample approach, with two samples per octave, to reduce the range over which the samples are stretched, reducing the distortion of the timbre.

It may be possible to create sounds that are still recognizable as acoustic instruments using a different synthesis technique, which also provides more timbral flexibility. One problem with the current approach, which provides a fixed set of timbres, is that MelodyMorph compositions eventually begin to sound alike. A system which gives the user some control over synthesis parameters would allow for a greater variety of timbres.

Providing help

I have generally introduced new users of MelodyMorph to the app with a brief demonstration. The question of how to help new users get started on their own is still an open design problem. In the most recent version, I have added a "help" button, which reveals a window showing a sequence of help pages. These each display a short looping animated cartoon, demonstrating a particular feature by showing one or two hands interacting with an iPad. This system does not seem adequate by itself to help people get started. Other possibilities to experiment with for the future include an animated or live-action intro and tutorial video, a

translucent overlay showing descriptions of different controls on the screen, and context-sensitive help that appears when the user takes certain actions.

Related Work

Creative tools for musical novices have been proliferating in recent years. I will focus here on tools that enable learners to create persistent musical artifacts, as in MelodyMorph. These fall into two categories: visual composition tools and tools for instrument-making with touch screens.

Visual Composition Tools

A sampling of visual music composition tools might include those using standard multi-track time and pitch representations such as Apple's GarageBand; grid based sequencers like the Tenori-On and its derivatives; modular tools for constructing music from chunks, including GarageBand's loop library and Impromptu (described in chapter one); and musical painting software like Morton Subotnick's Pitch Painter and Hyperscore (also described in chapter one). In spite of the variety of representations across these tools, they all have fixed mappings between space and musical parameters, unlike MelodyMorph. They do provide a variety of flexible features of other kinds, such as a huge range of synthesized and recorded instrument sounds, tools for manipulating sounds, automatic tuning and harmonization, etc.
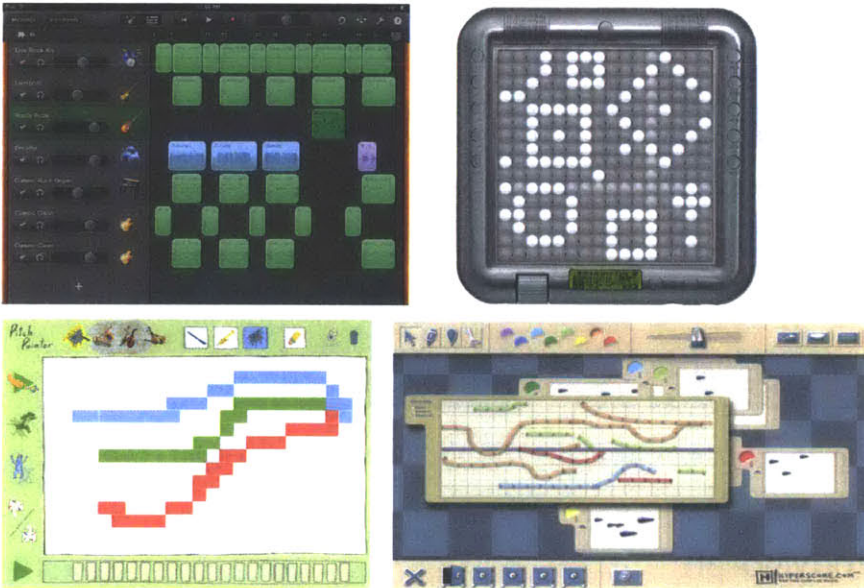
Figure 18: Visual composition tools. Top left: Apple's Garageband editor. Top right: The Tenori-on sequencer. Bottom right: Hyperscore. Bottom left: Morton Subotnick's Pitch Painter.

Instrument Making Tools

There is a smaller family of tools for making your own musical instrument. Apps for users

making their own screen-based instruments include Tachyon (combine and transform synthesized instrument timbres), Orphion (arrange circular pads for playing different pitches on a unique instrument sound), and Beatsurfing (create MIDI controllers by positioning shapes on the screen). Of these, Beatsurfing has the most in common with MelodyMorph, because it allows you to arrange various sound-triggering shapes on the screen in spatial arrangements of your choosing. From the complexity of its interface, It appears to be intended for people already familiar with professional music production tools, rather than a music education context. It does not, for example, play any sounds on its own: it can only be used to send MIDI to control another app or software synthesizer.
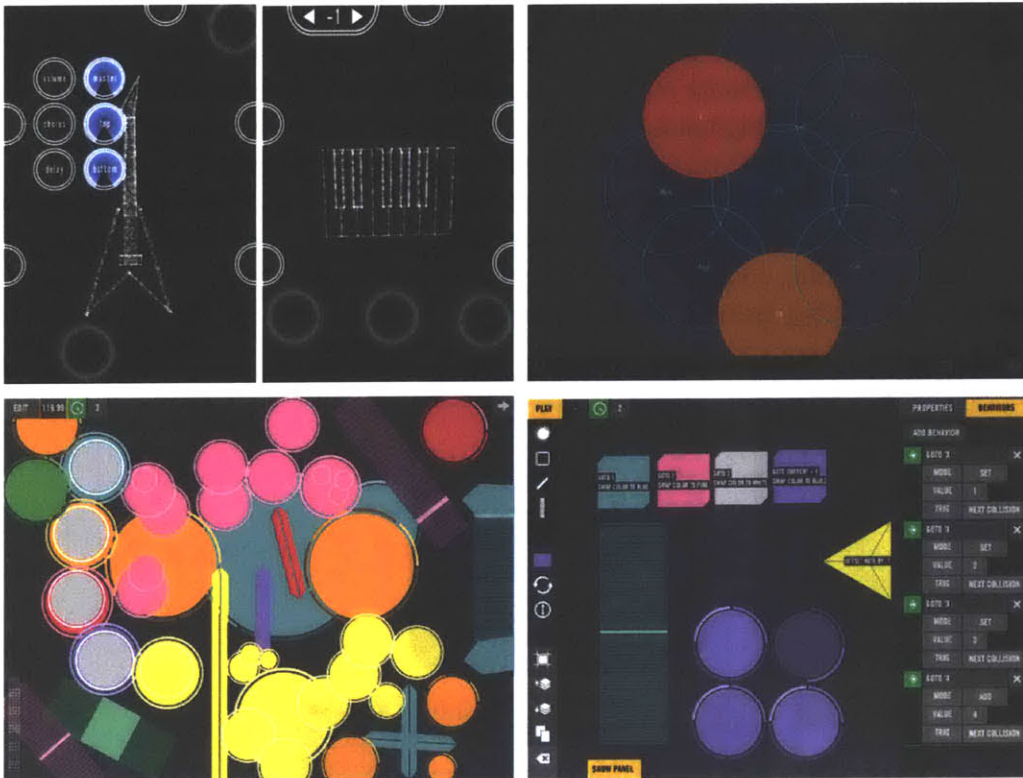
Figure 19: Touch-screen instrument-making apps. Top left: Tachyon. Top right: Orphion. Bottom left and right: Beatsurfing.

The Future of MelodyMorph

Future features

I have numerous other ideas for additions to the musical functionality of MelodyMorph. I could add MIDI output, so that the app could control other music software, making available a much wider range of sounds and effects. I could add a sampler, that allows the user to record their own sound, and then play it like the other instruments in MelodyMorph, transforming the sound to play it on different pitches. I could also broaden the set of instruments currently

available in the app, including percussion sounds as well. Ideally I would also be able to enhance the expressivity of the app by using sensor measurements to control the effect of hand gestures on musical events, such as the loudness of a note depending on how hard you tap it (early experiments with this feature have proven too unreliable).

Adding to the editing features, I could create additional ways to transform existing objects in pitch and time. I could add explicit support for specific musical scales, harmonies, and transposition processes. I could create a more hierarchical editing system, so that users could open up recorder bells and modify their contents. I could add features that help users play patterns that synchronize well in time (accurately timing the combination of multiple melodies is a challenge in using the software in its current form to make complex compositions). One idea for supporting better synchronization would be to introduce an object that takes the form of a line that sweeps across a region the canvas, triggering any bells as it touches them. This "sweeper" object could be set to run once, or to loop this sweeping movement. It could also make a rotary movement, sweeping around a circle.

Sharing

At the time of writing, the development of MelodyMorph's sharing system is at an early stage. The sharing server is implemented in Python using the Django web framework. The app can upload projects (including thumbnail images) to this server, browse the thumbnails in a grid view, and download and open them. Some additional features will be required before a minimal version of the sharing system can be included in a release of the app. It will need a way to create user accounts, and basic features for sorting and searching through morphs on the server. Ultimately, I would like to build up a community site through which users can freely share, communicate, and collaborate on MelodyMorph projects. This would involve adding social networking features that support various ways for people learn from and inspire each other. Based on my experience with users having a wide range of musical expertise, I can imagine users engaging with this community in different ways. I hope that novice users will be inspired by and learn from experts, and that experts and educators will develop content for novices as well as experimenting with ideas that push the boundaries of their own musical practice.

Chapter 3:
MelodyMorph Case Studies

In this chapter I present case studies of musical tinkering with MelodyMorph. After describing my research methods, I dive into the details of my work with middle school students making music with the app. Especially focusing on the story of one student, with some short studies of others, I observe the patterns of experimenting, expressing, and connecting. I also analyze tinkering processes in terms of loops and maps, looking at how loops at the different levels play out, and what types of tinkering maps the students make. Finally, I step back and describe how the creative potential of MelodyMorph derives from the intertwining of loops and maps.

Research Methods

Study Setup

I collaborated on a classroom study of MelodyMorph with Gena Greher, a professor of music education at UMass Lowell, along with four graduate and undergraduate music education students mentored by her. We partnered with a local public middle school, and worked with a music teacher there to run a class using the MelodyMorph app with fifth grade students (approximately ten years old). The classes took place during the regular school day, once a week, and were each about 45 minutes long. We ran a total of ten sessions. We worked with two groups of about 18 students each, seeing one group six times, and the other four times.

The public middle school we worked with is a Title 1 school, meaning that it receives supplemental funding from a federal program to assist students who are in poverty or otherwise at-risk. The state's classification puts it among the lowest performing 20% of its schools. The school serves 679 students in grades 5 through 8. Over 90% of the students in the school live in low-income households, with over 80% qualifying for free lunch and 10% for reduced-price lunch. Across the school the students are approximately 60% Asian, 25% Hispanic, 8% White, and 4% African American. The first language of 65% of the students is not English, and 42% are classified as English language learners.

We obtained permission to conduct our research study from the parents or guardians of the students using an informed consent form. We also obtained their consent to publish videos and photos of the students. We used an assent form to obtain permission from the students themselves. These agreements allowed us to bring the iPad app into the classroom for the students to use, to interact with the students verbally, to collect their digital creations, and to record these events on video. The video only recorded the participants' hands interacting with the technology, and not their faces. Properly de-identifying the video data also required removing images of hands with unique or unusual characteristics.

The researchers met on two occasions in advance of the classroom sessions, to get everyone familiarized with the app and to brainstorm activity ideas. In the classroom, I led the introductions to each session, with support from the music education students. Most of the time during each session was spent with the students working with the app, and researchers moving between groups, observing and assisting. The researchers also shared responsibility for recording video of our interactions with the students for later analysis, using small handheld cameras. Professor Greher and the classroom teacher hosting us stayed in the background.

Pedagogy

In this study, we used a facilitation style inspired by Critical Exploration (Duckworth, 1996). This approach involves working to understand what learners understand, and helping them ask and answer their own questions. The classroom was structured with the learners grouped at tables with two or three students each, sharing one iPad per table. We engaged the whole group in discussion occasionally, but most of the time was spent working in groups. The groups shared their creations with the whole class at the end of most sessions. We designed a sequence of activities, planning them as we went, based on our reflections after each session, and on new ideas as they arose. We started with an open-ended scavenger hunt for software features, and then progressed through musical puzzle activities, a musical cartoon project, a musical story, and finally an open-ended project developed over two sessions (see previous chapter for detailed descriptions of these activities).

Data Analysis

To analyze the data, consisting of video from the classroom sessions, I used a simple qualitative coding process. The process I used is inspired by grounded theory (e.g. Corbin and Strauss, 1990), applying some of its concepts in an informal fashion. My goal was to gradually make sense of the video data by finding recurring patterns and commonalities across the sessions. I labelled and clustered these groups of common occurrences. This process helped me to see the data at a higher level of abstraction, above the complexities of the many individual interactions. I used the resulting clusters of tags to select material for the case study analysis, which reassembles many brief observations into larger narratives.

My method draws only on some aspects of grounded theory methodology, leaving others out. My process of labeling video data and clustering the labels is inspired by its more formal process of developing concepts and categories (Corbin and Strauss, 1990). A rigorous application of grounded theory involves the continuous development of these theoretical constructs throughout the data collection process, and allows them to shape the sampling process by which subjects and contexts are chosen. I started in on my analysis during the time we were collecting data, but performed most of it afterward. I did not intentionally use the analysis to shape the sampling process (though I did notice interesting phenomena, and these biased my choices as I recorded certain students more than others). I made some use of comparisons, but generally did not use other elements of grounded theory methodology such as writing theoretical memos or incorporating broader structural conditions.

First, I made a coarse pass through the video clips we recorded in each session, labeling each clip with a short description of its contents to help me recall interesting events. These descriptions sometimes used a name for the student's project ("alien musical invasion of earth" or "my singing monsters"), sometimes included things the students said ("favorite franchises" or "oh that's so sad"), and sometimes described the students' actions ("[student and other student] work on puzzle" or "[student] presents solution two ways"). I also highlighted the names of those clips which included especially clear and detailed interactions.

Next, for a subset of the clips (drawn from the highlighted ones) I transcribed the speech and musical events. Then I made iterative passes through these transcriptions, applying tags. Each tag is a label for the action in a short segment of the video, such as a short spoken exchange, musical gesture, or software interaction. For example, when a student describes the narrative events in their musical cartoon to a facilitator, I apply the tag "storytelling." When they poke around in the palette trying out different notes, I apply the tag "exploring." In successive passes through the transcription I refined my definition of each tag, sometimes splitting them apart or combining them together. The result was a total of 11 tags. Finally, I clustered the tags into three higher level categories: experimenting, expressing, and connecting (see table 1). I used these tags and tag clusters as I constructed the case study narratives. I used them to select a set of interesting moments that ranges across the different tags, and to help describe these moments.

EXPERIMENTING

Anchoring   Reusing prior knowledge

Exploring   Trying new things without an explicit goal

Iterating   Trying different strategies to achieve a goal

Transforming   Making a change to an existing construction

EXPRESSING

Formalizing   Describing musical properties and relationships in words

Mapping   Using spatial relationships to show musical ideas

Emoting   Communicating emotion by performing, composing, or drawing

Storytelling   Describing, performing, or drawing narrative events

CONNECTING

Positioning   Enacting a social role

Collaborating   Asking questions and making suggestions

Co-creating   Making something together

Table 1: Tags applied to individual events in the video recordings.

A note on pedagogy, puzzles, and play

Before I begin the case study narratives, I will describe a challenge that arose as we designed the MelodyMorph activities to use in the classroom. In retrospect, I realize that the problem was caused by a mismatch between me and my collaborators (the music education students who co-taught with me), in the way we were thinking about learning and learning goals. The design of MelodyMorph from the beginning has been focused on open-ended creative music-making. But as we began to think about using it in a classroom with children, we started to focus on using it to make puzzles, which involve a different kind of learning experience.

Confronted with MelodyMorph for the first time, my undergraduate and graduate student collaborators used their existing musical knowledge to experiment with harmonies and melodies, and quickly made a variety of complex constructions. But as soon as we started talking about the middle school classroom, they reimagined their own constructions as things they could put in front of the children. One of these became a puzzle in which the child must

fill in missing notes along a path. Another became a puzzle that required matching different pre-made chords with the parts of a song.

These puzzles are exercises that seem inspired by a traditional notion of music pedagogy, with specific specific learning goals about melody and harmony. Puzzles can be answered correctly or incorrectly, so it is clear if each student has gotten the right answer, and apparently learned the concept. This pedagogical approach of course conflicts with a more constructionist approach, in which students have the freedom to stumble upon powerful musical ideas as they play and create according to their own interests. I nonetheless followed along with the idea of introducing the app using puzzles. I did this in part so that I would be incorporating my collaborators' ideas rather than totally imposing my own. But I was also probably influenced by my own unconscious fear that middle school students would not know what to do with the app, and the feeling that we would need to provide highly structured activities. Perhaps also we were all influenced by the feeling that we would need to create highly convergent activities, in which the students arrive at a particular answer, in order to fit into the curricular expectations of a school classroom.

In our first few sessions in the classroom, we asked the students to solve a particular musical puzzle using the app. One of these was to recreate the bass line from the song "Stand By Me" (shown in chapter 2). At the time, I felt frustrated, because the activities did not go very well. Some students found it too easy, others found it too hard, and none seemed very creatively inspired by the puzzles. I also saw that some of the students veered away from the puzzle tasks and started creating on their own.

After these initial sessions, I was able to shift our approach toward more and more open-ended creative activities. By the last sessions, we had the students working on final projects of their own choosing. My collaborators may at first have felt uncomfortable with this open-endedness, which it made it difficult to identify, at first, what the students were learning, and how it met their own pedagogical goals. But by the end of our sessions together, I think they saw the value in this approach. One of the music education students wrote, answering a question about what he had learned during the semester:

> *"I learned that it is just as important to let students work on their own to attempt to solve a problem as it is to teach them how to solve a problem. A student will get more out of the situation and gain more understanding when left to determine the answers on their own."*

As to what he would do differently the next time, he wrote:

> *"I would attempt to do even less. Give guidance but allow the students to create their own activities and projects with the Melody Morph program."*

And as an example of what the students learned:

"A student demonstrated on his own how the same melody can be played in different octaves or tempos to create a different mood. The student was given the opportunity to explore with the program and made this discovery on his own."

These comments, along with others I heard in person, and in the papers the music education students wrote about their teaching experience, lead me to believe that we successfully made a shift toward a more open-ended pedagogical approach. In the future, rather than starting out with musical puzzles, I can imagine weaving them into a mix of more open-ended creative activities. Puzzles may also be one useful structure (among many) to help beginners get started in a more solitary context, if they are introduced to the app without any in-person support.

Case Study: Sandy's Favorite Franchises

Sandy was one of the fifth graders who used MelodyMorph with us over the course of six weekly class periods. He caught my attention right away with the surprising sophistication of his ideas and his creations, and his idiosyncratic ways of expressing himself. We learned partway through the study that he has Asperger's syndrome. In this case study I describe his initial explorations, a musical cartoon he created in collaboration with another student, and his reconstruction of melodies from a video game. In between these descriptions I comment on how they illustrate musical tinkering, in the forms of experimenting, expressing, and connecting.

A favorite franchise

During our first session, Sandy began exploring the app and creating arrangements of bells on the screen. First he created a row of five vibraphone bells (triangles), left to right, in the order of a C major scale: C D E F G.

Figure 1: Five vibraphone bells in the order of a C major scale: C D E F G.

Next, Sandy proudly announced: "this is from one of my favorite franchises," and played a short melody using three of the bells in his row, the C, E and

Figure 2: The first few notes of the "Super Mario Brothers" video game theme, played by Sandy on his row of bells.

This melody is the very beginning of the classic theme from the video game "Super Mario Brothers." As I watched, he added a row of five "pluck" instrument notes (with a starburst shape) over the vibes notes, with the same ordering in a scale (see figure 3). Then he played the Mario theme again, using two fingers to play it on both instruments at once. This combination created a new timbre, combining the pluck and vibraphone sounds. Finally, he added a third row, this time adding the electric piano instrument, and played the same theme on all three instruments. To my ear, this combination of the three timbres brings it closer to the original 8-bit video game music sound.
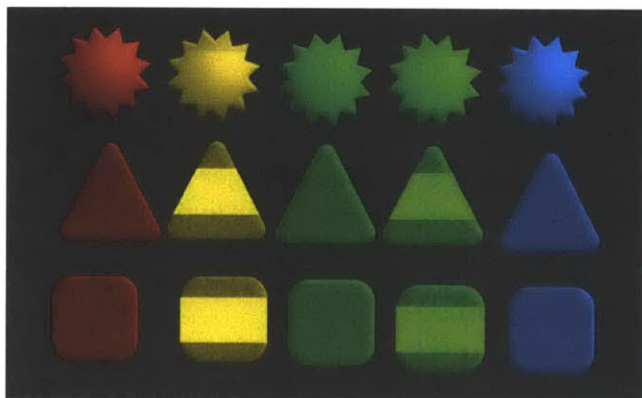
Figure 3: Three rows of five notes in a major scale, C D E F G. Each row contains bells for a different instrument sound (from the top: pluck, vibraphone, electric piano).

Notes: Experiments, starting with an anchor
In this first session we had given the students a very open-ended task, to find and report on MelodyMorph's features. Sandy took advantage of this freedom to dive into his own explorations in a way that demonstrates the four experimentation processes: anchoring, exploring, iterating, and transforming.

First, Sandy confronted this new and unfamiliar app by anchoring himself in something familiar, using his prior knowledge to make something simple that he could understand. He used his knowledge of the piano keyboard and of the major scale to assemble a row of five

notes on the screen (figure 1). This row also mimics the arrangement of the bells as they sit in the palette at the top of the screen (see screenshot in the previous chapter, figure 4). Rather than launching into something entirely new, Sandy used his existing knowledge to build a foundation to anchor his experiments.

Next, an exploration process led him to use the same row of five notes to play a familiar video game melody. He did not plan to make an arrangement that would allow him to play this melody, which uses only three of the five notes in his row. I did not observe this part of the process, but my guess is that he stumbled onto it by accident as he played around with the row of notes without a specific goal. It is also possible that first he decided to play this video game melody, and constructed the row of bells for that purpose.

Sandy was curious about the instrument sounds in the app, and did some iterative experiments in combining them. Rather than building something new, he anchored these experiments by re-using both the same row of five notes and the same video game melody. He added new rows of notes using two of the other instrument sounds, so that he could play the melody on first two and then three instruments simultaneously, and compare the resulting sounds. This iterative experiment with timbre combinations is also a simple example of transforming: he took the same row of notes and reconstructed it, changing only the instrument sound.

In this short episode, we see small examples of all four of the experimentation processes. As we progress, Sandy engages more deeply in experimentation in each of these ways.

Path variations

Next Sandy tried out the path-making mode in MelodyMorph. With the same triple row of bells on his screen, he enabled the path-making mode, and drew a path through them with his finger, leaving a line on the screen. By pressing the path player bell at the beginning of the line, he could then trigger the path to play back, causing a little triangle to ride along the bells, triggering the bells again with the same pattern in space and time as he traced with his finger.

For this first path, he traced his finger slowly through each column of three bells of the same pitch, starting in the middle on the E, then moving to the C, then passed through each of the other columns once, and returned to the C again to end (see figure 4). The Mario Brothers melody he had played earlier also begins with an E followed by a C, so perhaps he was recreating part of his earlier tapping gesture as a sliding movement. Once he had proceeded this far with the sliding gesture, he continued it with its own logic: unlike the tapping movements he used earlier, the sliding path motion lends itself to an alternating movement, up and down through the columns of notes.
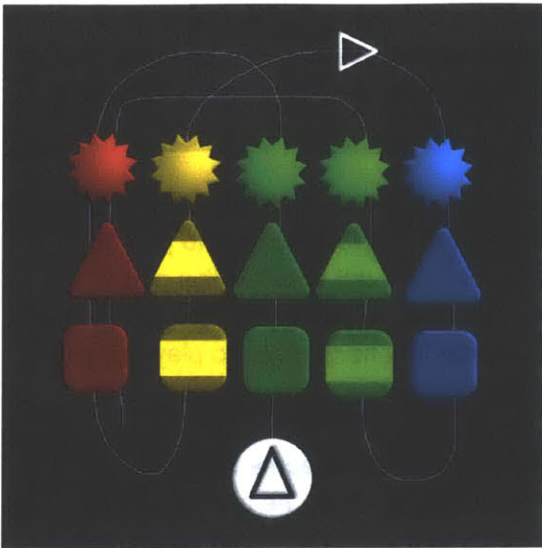
Figure 4: Sandy's first path. The path begins at the bottom center of the image at the path player bell, which can be tapped to trigger the path. The path head (white triangle at top right) moves along the path, triggering the bells.

A few minutes later in our 45-minute session, I returned to find a new configuration of bells on the screen. This new arrangement, of eight bells in two rows of four, was probably put on the canvas by the other student at the table (though it could have been created by Sandy himself). Sandy had drawn a path through it that played all eight notes, in alternating movements up and down through the columns of two (see figure 5a). Sandy announced "we created this really catchy tune," and tapped the path player bell to play it for us. Here we can see the same alternating movement as in the previous path, sliding up and down. This time he performed the sliding in a precise rhythmic way. The rhythmic pattern is made up of quick pairs of notes, for each vertical slide up or down. The resulting rhythm is a sequence of triplets. (Of course, the standard notation shown in figure 6 uses typical triplet groupings. In this case the visual groupings are different from the ones we perceive, which correspond to the pair of notes in each column). Sandy also now knew that by connecting the path back to its beginning, he could make it trigger itself in an infinite loop. By moving quickly back to the path player, he kept the triplet timing, so that it accurately looped in 4/4 time.

Next Sandy said "now let's see if we can do that backwards." He created a new path that traversed the bells in the same shape, but in the opposite direction (figure 5b). This path produced a melody with the same rhythmic pattern, but with the notes in reverse order. He observed that the melody was "still catchy." Finally, having explored the up-and-down path shape in two ways, he created a new variation. Without moving any of the bells, he created a melody that sounded quite different from the first two, by making a path that moved horizontally through them in a rectangular shape (figure 5c). This new path produced both a different melody and a new, irregular rhythm.
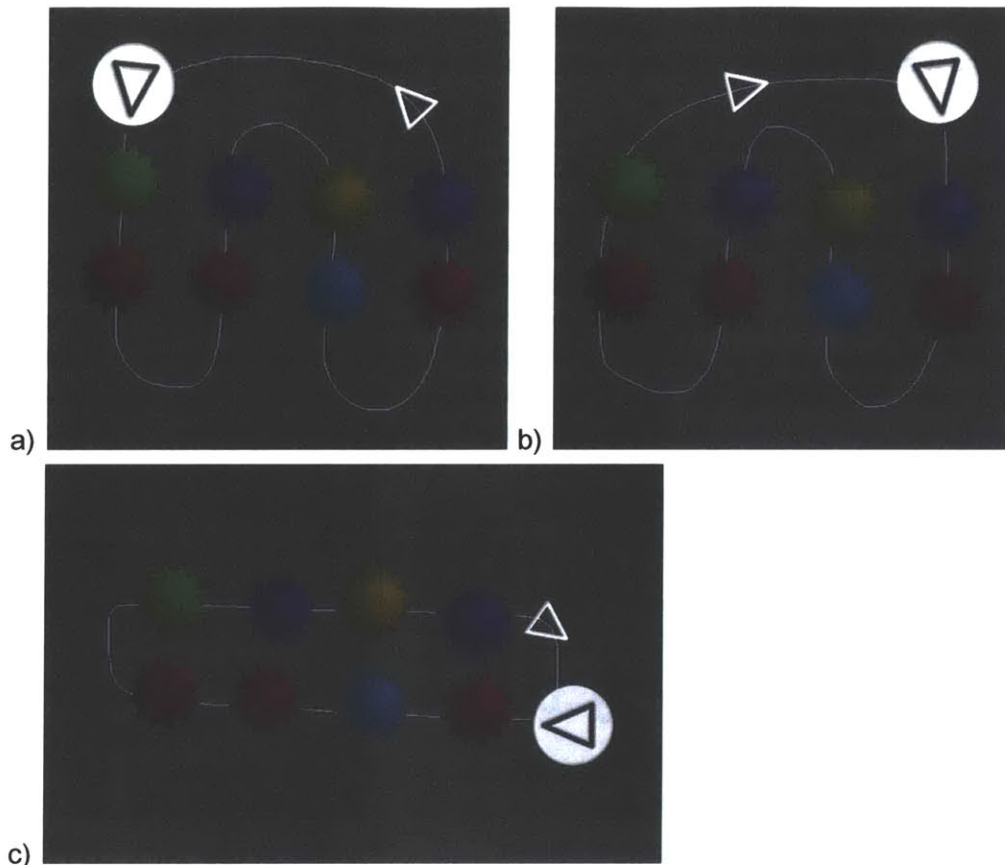
Figure 5: Variations on a path. A path plays a sequence of 8 pluck notes a) in its initial version, b) in the same shape but in reverse and c) in a rectangular shape.

Figure 6: A "catchy tune." This is the melody played by the path in figure 5a.

Notes: Experimenting with time and space

More experimentation appears in Sandy's variations on a path. Again we see him anchoring his explorations in familiar material. This time he is anchoring himself in his own prior work, building on the configurations he has already made on the screen. We also see a clear example of exploration, in his three variations on a path: forward, backward, and rectangular. He didn't create these paths with the goal of generating a particular melody. Instead, he was exploring the space of possibilities created by the path tool, evaluating the results as he went

("catchy" and "still catchy"). And he uses transformation in a complex way, taking the same set of pitches and making three very different melodies from them.

Canon experiments

Later in the first session, we demonstrated the recording feature. Sandy by this point had a new arrangement of bells on his screen, this time a full major scale divided into two rows (figure 7a). He used it to record a melody into a recorder bell (figure 7b). This melody proceeds in a way that's similar to his earlier "backwards" melody: moving his hand right to left, alternating up and down. Here he starts with the high C at the bottom right, moves up to the E, left to the D, down to the B, and so on (see figures 7b and 7c). The pattern is a steady series of eighth notes. He triggered this recording repeatedly, with accurate enough timing to keep the pattern going as a continuous sequence of eighth notes in 4/4 time.

a)

b)

c)

d)
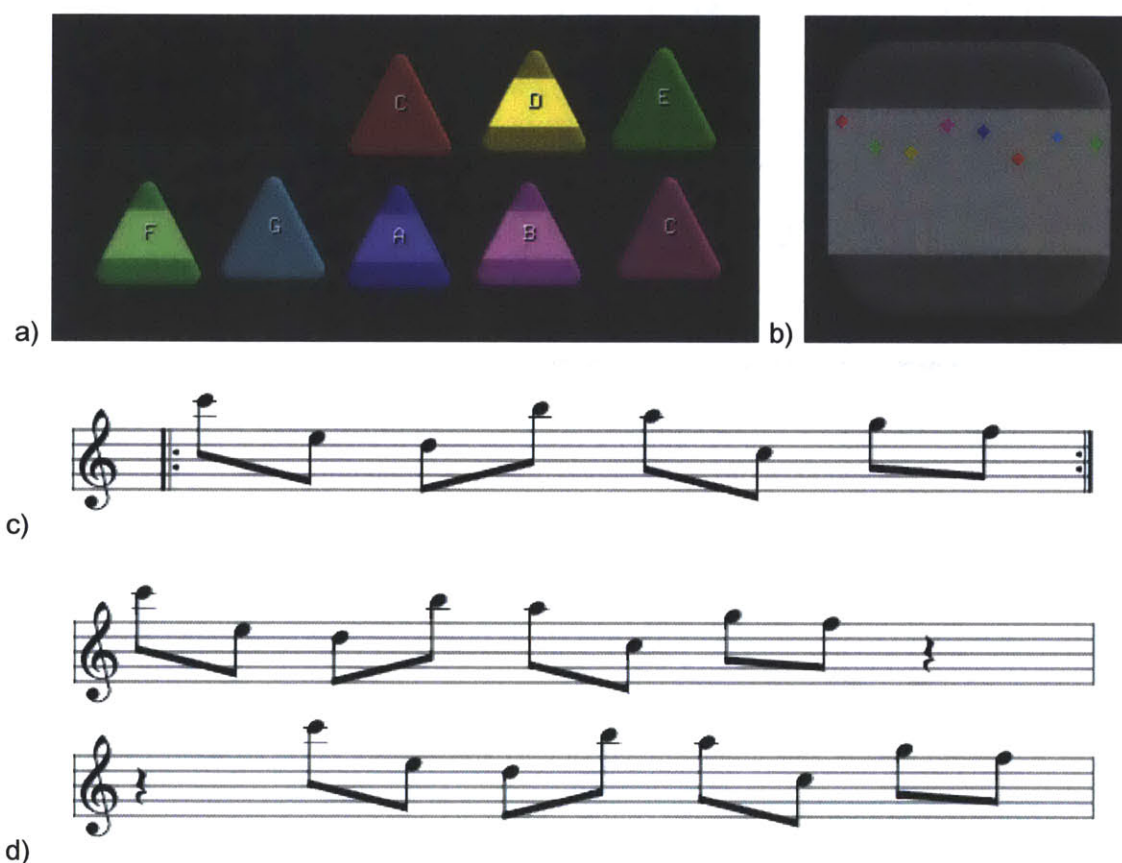
Figure 7: A canon. a) A set of vibraphone bells used to make a melody (pitch labels shown here for clarity) b) a recorder bell containing the melody c) the same melody in standard notation d) the melody in canon with itself, the second line being the same melody delayed by two notes.

I asked Sandy what he was noticing, and he said: "There's this different screen that shows like an invisible staff with visible notes on it." He's referring to the recorder bell, which does indeed have a kind of notation on it, in a style typically referred to as "melodic contour" notation. Because the notes are arranged with pitch on the Y axis and time on the X axis, in a way that is similar to standard notation, his description of it as an invisible staff is insightful. The fact that he called it an "invisible staff" suggests that he may not have seen this type of melodic contour notation before, and that he understood it by constructing a metaphor relating it to standard notation.

I noticed the next time he triggered the recorder bell that he was playing along with it, tapping the original notes as it played. I suggested a "trick," which is to record the recorder bell, and add more notes as it plays (a bit like multi-track recording, or overdubbing). I said, "If you set it to recording again, and you play the one you recorded, you can, like, add to it. So you can make new layers."

Instead of overdubbing, as I suggested, he did something simpler: he simply pressed the record button, and then triggered the existing recorder bell. This created a new copy of the recorder bell. It was a duplicate, so he now had two identical recorder bells. First, he played them at the same time. I asked "what if you played them, instead of the same time, at different times?" He immediately played them one note apart (starting one, then starting the second one after one note had played). In music theory terms, this is a canon. Then he tried it again, playing them two notes apart.

Finally, he made a path that slides through the two recordings so they start only about a sixteenth note apart. This created a sound a bit like an echo, with each note triggering twice in rapid succession. The path continued through the bells, and returned to trigger itself again in a loop. While a bit chaotic sounding, this last creation was significant because it combined in one artifact several of the ideas he had played with separately during this session: the recording, a variation on the idea of a canon, and the path.

Notes: Experimenting with new combinations

In this episode, Sandy tinkers in yet more sophisticated ways. He anchors himself by reusing the skills he has just learned, making a new path with a similar structure to his earlier one. He uses the idea of transformation once again to create different canons, by trying three different intervals of time to separate the voices: one note apart, two notes, and a smaller fractional amount. He also makes a more complex transformation of his musical material by combining it in a new way, putting together path, recorder bell, and canon.

He also does what I call "formalizing," which is describing a musical phenomenon in words. In this case, he uses a metaphor to help him understand something new. He describes the melodic contour notation on the recorder bell as an "invisible staff." This is also an example of anchoring, because he is referring to a notation with which he is already familiar.

Sandy and Daniel's musical cartoon

In our third session with this same group, we began by showing the students some examples of "musical cartoons," which we had created by drawing characters on the screen with speech balloons, and constructing melodies to put inside them. One example showed a bird with a trilling melody in the speech balloon. Another example showed a mother elephant, with a low, plodding, piano melody, and a baby elephant replying with a similar melody in a higher register. For this session we gave each group a template project, containing rows of notes in each of four different scales, all in the key of C: major, minor, harmonic minor, and whole tone. We told the students that they could use notes from these scales to create melodies for their cartoons, but without explanation of what the different scales might be useful for. Then we brainstormed briefly as a group about the kinds of cartoons they might make.

Sandy spent this session working with another student named Daniel, making a story about two characters. At an early stage in the process, I saw that Sandy had drawn a smiley face with a speech balloon. He had worked out a melody on a row of bells that formed the upper part of a C major scale, and put the row into the speech balloon (see figure 8).

Figure 8: A smiley guy and with his melody in a speech balloon. The pitches in the speech balloon are, from left to right: F G A B C' (the apostrophe indicates that the C is in the upper octave, i.e. the notes form the upper part of a C major scale).

Next he said "now we have to record it," and turned on the recorder and played the melody shown in figure 8. Later I said "so maybe we'll give the other guy a turn?" and Sandy said, "Now it's Daniel's turn." Daniel drew another face (figure 9) to the left of Sandy's smiley face. Sandy noted that "this guy has spiky hair," perhaps in order to clarify that these are two

different characters. Daniel finished the drawing by adding a frowning mouth and a speech balloon. Then he began playing the minor scale notes, and pulled a row of minor scale notes into the speech balloon. Note that the row is not exactly in the order of a scale, since the Bb comes before the Ab (see figure 9).
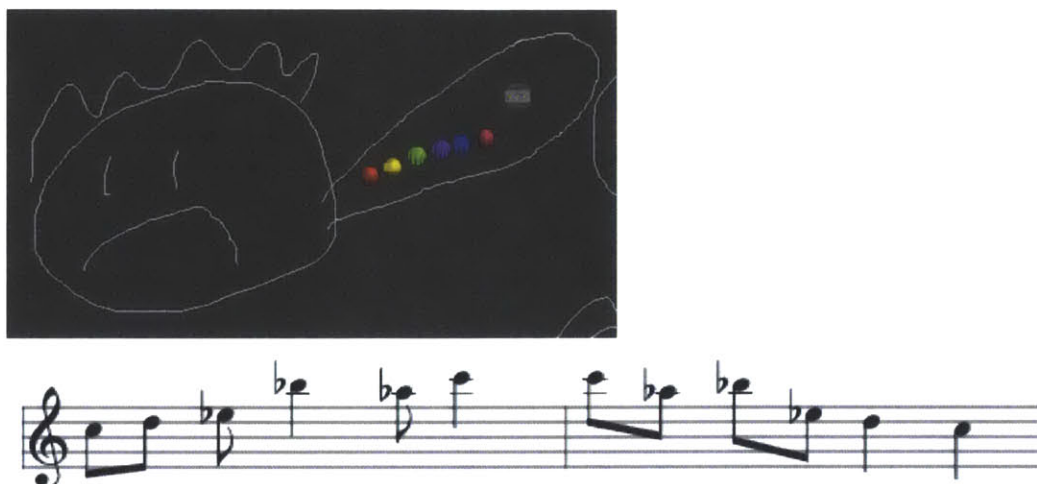
Figure 9: The spiky guy and his melody in a speech balloon. The pitches of the bells are, left to right: C D Eb Bb Ab C'.

Initially, the row contained this sequence of five pitches: C D Eb Bb Ab. He played up and down this row of bells, searching for what might come next. He tried the top note of the scale in the palette, C', then the D and the Eb. Next he played left to right along his row, and moved back to conclude on the Eb. Sandy asked "do you like your song?" and Daniel replied "no not yet." I asked "what's missing?"

This time Daniel played his sequence from left to right and concluded with the C' in the palette, then dragged it down to the end of the sequence. Sandy said "now we can record it" and pressed the record button. Daniel recorded the melody in the figure, which moves across the row left to right, and then from goes back, from right to left.

Next, Sandy triggered Daniel's minor melody recording, explaining the story so far: "so this guy is kinda sad because he doesn't have the video game that I have." Then he triggered his own recorded major scale melody, saying "and I'm playing my video game."

Later, we returned to the table and Sandy gave us an update. On the screen they had two scenes. The first scene is what we've seen already: Daniel's spiky hair character with a minor melody, and Sandy's smiley character with a major melody. In the next scene, Sandy's character has a melody in which Sandy says he is "notificating [i.e. notifying] this guy [Daniel's character] that he can play with him." This speech balloon has just two bells: A and C'. The

recorded melody is shown in figure 10. To me this melody sounds like someone happily saying "okay! okay!"
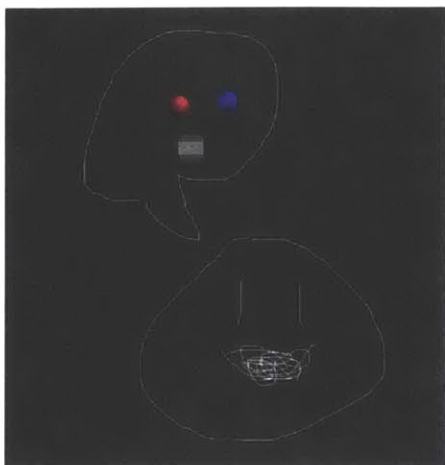
Figure 10: The smiley guy's second melody. Above, the smiley guy with the two notes for his melody in a speech balloon: C' A. Below, the melody notated.

Daniel then set about creating a response from his character. After searching around the minor scale palette, he moved to the major scale, and pulled into the new speech balloon a row of notes in this order: G G G A. After some iterative experiments playing this sequence of notes from left to right, he made a recording of this melody, and explained that the character is saying "thank you."
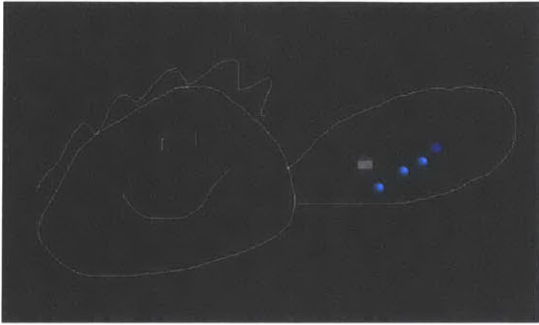
Figure 11: The spiky guy's second melody, for "thank you!" Above, the spiky guy with his speech balloon including these bells: G G G A. Below, the melody notated.

Later, Sandy said "actually I think these two things should be played at the same time because in the story they actually have the same time to start." He triggered the first two melodies' recorder bells simultaneously (i.e. the melodies in figures 8 and 9), with a somewhat dissonant result.

Sandy also added a fifth drawing, with a short melody in its speech balloon. The melody is a fragment of the bass line from the song "Stand by Me." The students in this music class had studied the song earlier in the semester, learning to play it on the piano. In our second MelodyMorph session with this group, we had asked the students to recreate the same bass line from the song. Its reappearance here (about three weeks later), in the form of an allusion, seems to be a kind of musical joke. This idea is reinforced by the fact that the face drawn for this melody is in a different style, with a smirking half-smile and raised eyebrows. It is tempting to think that Sandy's reference to the song connects its theme of loyal support to the story in his cartoon. It is also possible to see it simply as a slightly mischievous appropriation of classroom material.
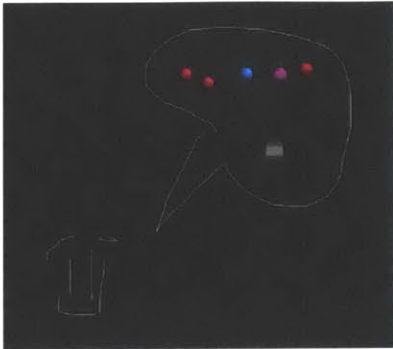
Figure 12: Character with a "Stand by Me" melody. The notes in the speech balloon, above, are C C G B C.

Near the end of the session, Sandy said to me "we've already got five scenes with two characters," and proudly played through the whole sequence of five melodies. Sandy had earlier explained "so this guy [the first spiky guy] is kinda sad because he doesn't have the video game that I have." The happy guy is Sandy, and in his first appearance he is happy because "I'm playing my video game." It's not really clear in what order the action takes place (the story seems to have changed around as Sandy and Daniel worked on it). At some point the happy guy tells the spiky guy that he can play the game ("notificating" him), with an "okay, okay!" melody. The spiky guy responds with a "thank you" melody. And finally there is a sort of joke at the end making reference to "Stand by Me."
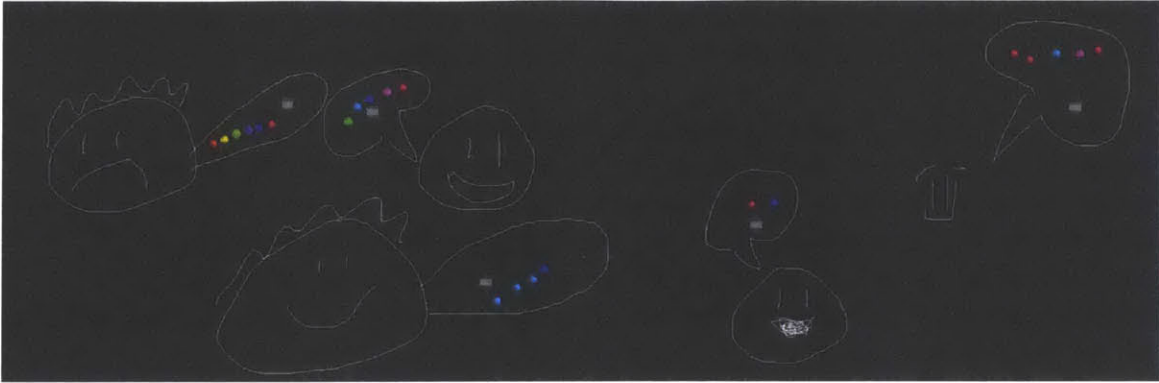
Figure 13: The entirety of Sandy and Daniel's musical cartoon. From left: first spiky guy, second spiky guy, first smiley guy, second smiley guy, and Stand by Me.

Notes: Collaborative storytelling and transformation

Sandy and Daniel collaborate in several ways as they create these two characters and their melodies, talking about turn taking, about their characters, and about their melodies. They create collaboratively as well, working sequentially to build up a story.

We can see a subtler form of collaboration by comparing their first two melodies (see figure 14): Daniel takes ideas from Sandy's melody, mimicking it in some ways, and transforming it in others. Both melodies are made up of two short phrases, both are played on a short row of notes, and both traverse a row of bells first in one direction and then in the other. But the melodies sound very different because Daniel's is minor and Sandy's is major. Daniel's melody traverses his row of notes in opposite directions from Sandy's: up then down, rather than down then up. Sandy's melody lands on the G, giving it an unresolved quality, while Daniel's melody ends on the C, giving it some finality.

Figure 14: The smiley guy's first melody (by Sandy), above, and the spiky guy's first melody (by Daniel), below.

These transformations are a kind of experimentation with musical form. We don't know if he has the vocabulary to describe them in these terms, but Daniel's iterations vary the melody's contour, the harmony it implies, and its resolution. Both students seem to continue with this experiment with musical form in the second part of the story: the "okay, okay" and "thank you"

73

melodies are both again made up of two short phrases, and the "thank you" melody also uses the pattern of traversing a row of bells in one direction and then the other.

It is worth noting that, though I attribute these relationships between melodies to a collaboration, it is possible that they are coincidental, or unintentional. I did not have the opportunity to document the creation process, or to ask the two students about them. It seems likely that the two were mimicking and responding to each others' compositions, so I speculate using that assumption. But it is also possible that the similarities result from common patterns of behavior, such as moving across a row of bells in one direction, and then back again. It is also possible that the similarities and differences occurred purely by chance.

Sandy's singing monsters

In our fifth and sixth sessions with the group, we gave the students the opportunity to develop their own projects using MelodyMorph. Sandy decided to recreate the music from a video game that he enjoys:

> "I'm trying to make a song from My Singing Monsters. And if you ever give this video to Big Blue Bubble [the game development company], they would be excited because they created My Singing Monsters, and now they know a fan… I'm creating a song that's stuck in my head after playing it."

My Singing Monsters is a game for mobile devices in which the player collects and breeds musical monsters, each of which sings its own looping melody. The monsters sing together, creating a complex texture of interweaving melodies. Sandy did not have access to the MelodyMorph app at home, and he did not have access to the game itself in our class, so his reconstructions of the music from the game are from memory. By our final session together, Sandy had a recorder bell containing a four-bar sequence layering together the melodies from three of the monsters: the "Mammot," the "Toe Jammer," and the "Oaktopus." He decided to add a melody from another monster, the "Furcorn."

Figure 15: Monsters from the game "My Singing Monsters." From left, the Mammot, the Toe Jammer, the Oaktopus, the Furcorn, and the Noggin (Images by Big Blue Bubble, inc.).

My conversation with him reveals a bit about Sandy's process of recreating music from this game.

Eric:	"Now what is going on, just in that recording you've already got? It seems like there's a lot going on."
Sandy:	"Well, see that blue blue light-blue blue purple red over there?"
Eric:	"Uh huh, in the middle."
Sandy:	"Well that's the bass line, and I know it because this is from "My Singing Monsters." And guess what? I looked in the game's data."

With the sequence of colors, Sandy was referring to the melodic contour notation on his recorder bell. This recorder bell contained his recording of the layered melodies of three monsters, the Mammot, Toe Jammer, and Oaktopus. He singled out one of the melodies, a bass line sung by the Oaktopus, by naming the colors of the dots in the notation. These colors match the colors on the bells, indicating their pitches.

What Sandy said next was quite surprising, and revealed the depth of his interest in both the game's music, and in recreating it using the app. He described to me in detail the process by which he "looked in the game's data," by opening up the "My Singing Monster" app's internal files on his computer, in order to access the recorded sounds. His words and tone of voice showed just how proud he was to have accomplished this.

Next I asked, "so what do you want to add to your singing monsters melody?" He said, "the Furcorn." I asked "the what?" At this point I did not yet have a detailed familiarity with the singing monsters in the game. Sandy replied, "the Furcorn, the monster who goes [sings the furcorn melody]."

Later, once he started working on the Furcorn melody, he explained: "this is the Furcorn and I want to record a new one with this [existing recording] and get them both synced." After singing the melody aloud, he created this row of bells:

C D E F

This row allowed him to play what he calls "the first part" of the melody:

Figure 16: Sandy's first attempt at the first part of the Furcorn melody.

Next he added a another row of notes, below the first, for the second part:

A' F' D' C'

And made a first attempt at playing it:

Figure 17: Sandy's first attempt at the second part of the Furcorn melody.

Not satisfied, he tried the last three notes again, on the lower pitches:
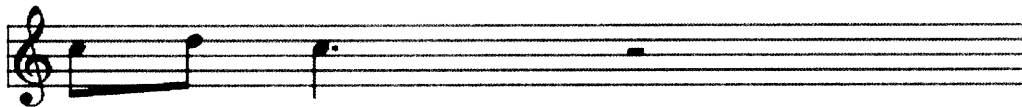
Figure 18: Sandy's next attempt at the last three notes of the Furcorn melody.

Then he rearranged the row, swapping the C and D bells. This changed the row from a descending scale to an ordering that reflects the structure of the melody:

A' F' C' D'

Of this second part of the Furcorn melody, he said: "this one was my best shot. I don't really know how to play the Furcorn on the keyboard. I only know this part [plays the first part]."

Next came the task of syncing the Furcorn melody with his existing four-bar recording of the other melodies. He already had created a path that triggers the four-bar recording, then comes back around to trigger it again at the right time to create a continuous loop. He says: "but this [the four-bar recording] is seven seconds, and the Furcorn's singing is fourteen seconds. So I guess we gotta use… repeation! [i.e. repetition]" What he had realized is that, unlike his four-bar-long performances of the Furcorn melody as he was experimenting, the actual Furcorn melody occurs over two loops through the four-bar cycle, making it eight bars long. In a first attempt at making a recording that syncs the two melodies, he made an error: "oops! I did it too fast, as if this [the four-bar melody] was not going to be repeated." On another attempt he succeeded in lining up the rhythms accurately, making an eight-bar recording.

Finally, as he prepared to present his project to the whole class, Sandy decided to add the sounds from yet another monster, the "Noggin" (see figure 15). The problem is that the Noggin makes drum sounds, which this version of MelodyMorph does not have. The monster is animated in the app, drumming with its hands on its own head, which Sandy enthusiastically acted out. Later, when Sandy presented his project for the whole class, he was able to trigger his loop, now containing four monsters' melodies layered together, and accompany it by playing the Noggin's rhythm on a bongo drum.

As he began his presentation for the class, Sandy said "raise your hand if you've heard of My Singing Monsters." There was a bit of confusion among the students, as most were not familiar with the game. When another student asked him to repeat the name, he said "My Singing Monsters. I know Danielle [another student] plays it." And then, addressing the teacher, he said "you paired me up with Danielle on the grand piano and we became friends."

Notes: Experimentation, expression, and social positioning

In this final example of a more complex creation from Sandy, we see rich experimentation, supported by expression. He experiments iteratively with melodies, attempting to recreate remembered music from the game. He tries different variations, rearranges the bells, and works on correctly lining up the melodies with each other in time. This experimentation is supported by an anchoring in his own remembered hearing of the melody, as well as his partial knowledge of how to play it on a piano. It can also be seen as a creative transformation of the game music, that makes it his own. The biggest difference is that Sandy's melodies are in a major key, while the music in the game is minor (see figure 19).

Figure 19: Sandy's version of the Furcorn melody compared with the original. Sandy's version, above, is in C major, and the game's version, below, is in C minor. The second part of the melody has the same contour but on a different set of scale degrees.

Sandy expresses himself both formally and emotionally in the process. He verbally expresses some complex ideas about the structure of the music, such as the two-part structure of the Furcorn melody, and the relationship between the two loop lengths (the other melodies are four bars long, and the Furcorn's is eight bars, though he describes their lengths in seconds, which I don't know how he measured). He expresses emotions in subtle ways throughout,

showing both his frustration and his enthusiasm. He most overtly expresses his excitement by drumming on his head, and then enthusiastically sharing his piece with the class, including his performance on a bongo drum.

Finally, as he stands up in front of the class to present his work, we see Sandy trying to assert his power and importance in the social environment of the music classroom. His first question, asking the students to raise their hands if they have heard of the game, makes use of a teacher-like pattern of interaction. When the students react with a bit of confusion, he tries a different tactic to earn their respect, explaining that he knows that one of the other students plays the game.

Sandy's story about how he became friends with another student at the grand piano is revealing as well. My guess is that, as a special needs student with Asperger's syndrome, Sandy is socially isolated. The fact the he emphasizes this one incident perhaps shows just how rarely he has a chance to make a connection with another student as an equal. On the other hand, it shows how musical creativity is one pathway for Sandy to create a more powerful social role for himself.

One of the graduate students facilitating the MelodyMorph sessions with us decided to focus, in her final paper for the semester, on her interactions with Sandy and the other special needs students in the class. She wrote:

> "When [Sandy] came into the room I told him that he could use my iPad, and he could create ANYTHING he wanted. His reaction made me realize how important this app really could be. He put his hands over his mouth, he had a huge smile and he jumped up and down, because he was so excited... Sandy was the first student to use external music making [i.e. playing acoustic instruments along with the iPad] in either of the 5th grade classes, and when he played he was able to create syncopation, and a clear meter for his melody. I showed him the video I took of him, and he was so excited to see and hear himself playing with the app. In that moment, I knew that I had achieved my research goal for the semester. I was able to use Melody Morph to help facilitate meaningful music learning and explore the potential for using this app with the special needs population."

Experimenting, Expressing, and Connecting in Sandy's Story

In the path we have traced through Sandy's work with MelodyMorph over the course of six class sessions, there are examples of each of the learning processes shown in table 1, above: experimenting, expressing, and connecting.

Sandy is an experimenter. We see him anchoring his experiments in familiar musical material, making use of themes from the video games Super Mario Brothers and My Singing Monsters. We also see him anchoring by building and reusing his existing constructions and techniques.

His explorations, semi-random wanderings in the palette or on a path, lead him to musical discoveries, such as familiar melodies. Through iterative experiments, he tries out different ways of doing things, evaluating the results. We can imagine that, internally, he asks himself questions like "is this tune also catchy?" and "does this sound like the music I remember from the game?" And he makes sophisticated use of transformation, using different paths to make different melodies with the same set of bells, and turning a melody into a canon in multiple ways.

Sandy expresses both ideas and emotions. He expresses his ideas verbally, telling us, for example, about the metaphor he uses to understand the "invisible staff" on the recorder bell. He expresses ideas in the app as well, using space to map out different timbres, or the arc of a melody. Sometimes he uses the ordering of a musical scale, and other times he arranges pitches according to their appearance in a melody. Sandy expresses emotion, in the form of pride in his own creations, as he shares them with the facilitators or with the whole class. He also expresses himself through stories, in a cartoon including melodies for happy and sad characters.

Finally, Sandy connects with other students in several ways. His conversation with other students shows collaboration on musical ideas, as does their co-creation of musical artifacts such as the multi-scene cartoon. Sandy also asserts his social position in the classroom by making teacher-like gestures in front of the class, and by emphasizing that he has made a friend who also plays the video game that he likes. He also positions himself socially in more subtle ways, by comparing his work to that of the other students, and trying to impress others with his accomplishments.

Loops and Maps in Sandy's Story

Sandy's numerous musical experiments take him through many tinkering loops, as he plays with timbre, melody, paths, and canons. We see complex context-loops in Sandy's collaborative experience with Daniel. As they work on a musical cartoon, a complex feedback loop of ideas and emotions guides their emerging creation. And in his Singing Monsters project, we see Sandy engaging in larger context-loops. The project itself is more complex, and requires a longer trajectory, but it also brings him into contact with the technology behind video game music, with the social complexities of sharing it with his peers, and even with combining MelodyMorph with an acoustic musical instrument. We can hope that this was not just a meaningful learning experience for Sandy, but one which drives a larger outer-loop, in which he develops a new understanding of himself in relation to music.

The tinkering maps that Sandy makes with MelodyMorph take several forms. As he experiments with paths, making a different melody with each shape, he shows the power of externalizing and transforming his ideas. The paths that he draws reflect (perhaps) the shapes of his ideas about musical structure, and also bring those ideas to life as melodies. He transforms his musical map in the cartoon project, placing bells sometimes in the order of a scale (as in the smiley guy melody), and other times in the order of melodies (as in the

"Standy By Me" melody). We see his musical thinking change in this way as he works on the Furcorn melody, shifting from scale order to melody order as he tries to puzzle it out.

Three Classroom Vignettes

Here I introduce three shorter cases studies from the same classroom study, about three projects: a marriage proposal, a morning melody, and the evolution of a looping instrument.

A Marriage Proposal

In our activity on "musical conversations," we asked the students to compose melodies that represent speech. Here's my initial conversation with a pair of girls. The girl named Jenny did the talking:

Eric:	*"What is your conversation going to be about?"*
Jenny:	*"We're trying to do like a marriage proposal."*
Eric:	*"So how do you make a musical marriage proposal?"*
Jenny:	*"Let's make one where they say yes, like really happy, like really high."*

Here Jenny played the top-most piano note in palette, a high C, several times

Eric:	*"So first you gotta have the person asking, though. So what does that sound like?"*
Jenny:	*"A guy... he's really nervous though."*
Eric:	*"What does that sound like in music? Maybe you could act it out first and make music from that."*

Jenny acted nervous for moment, making a short warbling, worried moan, and then laughed, embarrassed.

Eric:	*"That's good! there's a lot already in what you just did for a second there... When you're nervous, what does your voice sound like?"*
Jenny:	*"Shaky."*
Eric:	*"So could you make a shaky melody?"*

As we spoke, the other girl, Jessica, was tapping notes in the piano pallette. After I asked this question, Jessica played a quick triplet on the A. Then Jenny reached over and played a figure that imitated this triplet, then continued down the scale:

80

Figure 20: Jenny's shaky melody
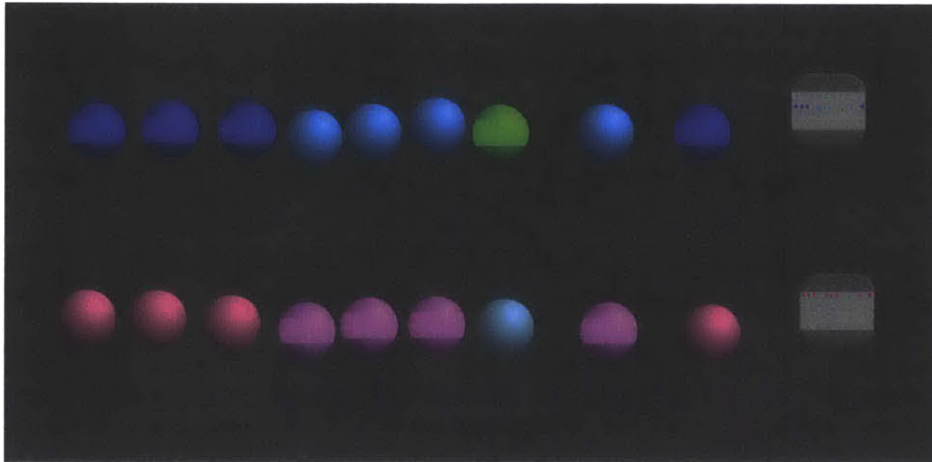
Figure 21: A musical marriage proposal. Above, the MelodyMorph screen shows, at the top, a row of bells for the guy asking, with pitches A A A G G G F G A. At the bottom of the screen, a row for the girl's reply, with pitches C C C B B B G B C. Below, the two melodies notated.

Later, Jenny and Jessica presented their musical conversation for the whole class.

Jenny: "So this conversation is about a guy proposing to a girl. And she said yes."

Eric: "So how did you choose the notes for the person asking?"

Jenny:	*"He's supposed to be nervous at the same time, even though he wants to do it, and then for the girl, she's excited and happy and says yes."*
Eric:	*"How did you show the nervousness with your melody?"*
Jenny:	*"Shaky, kind of... I tried to make it sound like low, but then like kind of trying not to show how he's nervous, but he is."*
Eric:	*"Were you thinking about the colors of the notes also on the screen when you chose them?"*
Jenny:	*"No [laughter]. Yeah. But no."*

Notes: Nervous and excited melodies

Jenny and Jessica translate their story of a marriage proposal into a MelodyMorph composition using musical ideas to communicate emotions. In their story, the guy proposing is nervous and his voice is shakey. They translate this shakiness into a pattern of repeated notes, transforming shakiness in the voice into a rhythmic tapping of the hand. The melody starts with an A repeating three times, then repeats this pattern a step down on a G, and ends with three notes ascending back to the first note, F G A. While the girls did not say so explicitly, they may have structured this melody around how the nervous proposal question would have sounded in words. Using a note for each syllable, and stretching the first two words with a nervous stutter to add extra syllables, it becomes: "wi-i-ll yo-o-ou ma-rry me?" The ascending melody for "marry me?" could represent the ascending voice pitch of a spoken question. It's also possible the that the students did not structure their melody using the sound of the spoken question, and that it derived instead from their intuitive tapping patterns and the cluster of bells that they dragged onto the screen.

The reply from the girl is a happy and excited yes. To represent these emotions our composers start with a high C, the highest pitch available in MelodyMorph. The melody for the reply mirrors the question in both its rhythmic structure and its melodic contour, descending with two groups of three repeated notes, then ascending back to the first note. The third-to-last note is a G, a scale degree lower than it would have been otherwise following the pattern from the first melody. With this change, the melody implies a typical chord pattern (a dominant-tonic cadence) that creates a strong sense of resolution or finality. We do not know if the girls were aware of this idea or if they stumbled on this melody by chance. It may have been suggested by a facilitator. It may even have come from the earlier experience the class had with the "Stand By Me" bass line, which contains a similar pattern.

It is worth noting some visual features of the girls' composition. The figure shows a reconstruction of the girls' screen, from what is visible in the video, so the placement of bells

is not exact. We can see that the spacing between bells increases in the last few bells of each row, but is otherwise consistent. This spacing indicates an approximate sense of the rhythm, rather than a detailed mapping between time and position. We also might wonder if the pitches were chosen partly for their colors, so that the guy's melody is mostly blue and the girl's mostly pink. The girls give an ambiguous answer. Perhaps the color scheme happened at first by accident, and then they chose to reinforce it.

A Morning Melody

In another activity we asked the students to create musical stories about different times of day. Ben's group composed melodies for the morning, going to school, going to sleep, and eating. In my conversation with Ben below, he explains some of the musical thinking behind the first two of these melodies.

Eric:	*"Tell me about your morning melody. How did you come up with it?"*
Ben:	*"I had to think of it like, I pressed all these notes [in the piano palette] and I seen which one describes morning the best."*
Eric:	*"And how did you know that those ones described morning the best?"*
Ben:	*"'Cause it didn't sound happy, but it didn't sound sad neither. It was in the middle."*
Eric:	*"Very interesting. And I see you have two sets of them. Why did you do that? it's like red yellow green, yellow red green."*
Ben:	*"I had it like that [he moves a bell to make the second group same as the first, red yellow green] but I noticed I had to mix it up cause it just sounded like the same whole thing, it just sounded all grumpy, but I tried to make it half, 'cause you just woke up, you're tired. But you just woke up and you have something to go to, you have to go to school."*
Eric:	*"So that changes it up, and you have your melody change."*

Figure 22: Ben's melodies for waking up in the morning, and going to school. The morning melody's pitches are C D E D C E. The school melody's pitches are A B B A' F'. Below, the first part of Ben's recorded morning melody.

Ben: *"But then for recess, when you're at recess you're loud. So we changed the notes to be loud and high"*

Eric: *"And how did you choose those notes?"*

Ben: *"We were looking through the high notes for the vibes, and we had to choose which ones described recess the best.*

Notes: Melody for a mix of emotions

Ben's explanation shows how he uses a melody to express his sense of the morning as a mix of emotions. He chooses notes that are not sad or happy but are "in the middle." I point out that his row contains two groups of bells that are slightly different, with pitches C D E and D C E. He explains that initially they were the same (i.e. C D E and C D E), but he had to "mix it up." The ability of MelodyMorph to make modifiable tinkering maps is shown here, when Ben quickly rearranges the bells to tell the story of how and why he changed his melody.

But how does Ben express a mix of emotions as a melody? He used the row of bells to record the melody notated in the figure above. He played the first phrase of the melody by playing the bells in order from left to right, then repeating the last note in the row, an E, and ending

with the second-to-last note in the row, a C. The second phrase is similar, but ends instead on the third-to-last note, a D. In the third and final phrase (not shown), he plays the bells again in order from left to right, ending on the last note in the row, an E. The sequence of three phrases have endings that feel less resolved each time, landing on a C, then a D, then an E. To my ear, the melody as a whole expresses a transition from a more static, sleepy state (starting on C and quickly resolving back to C), to a more energetic, unresolved one, that ends feeling like it has "something to go to." From Ben's description, we might imagine that his melody is made up of two separate parts, one less and one more energetic, using the two groups of three bells. Instead, the melody moves across the whole set of six bells, creating a progression by changing the ending of each phrase.

Evolution of a Looping Instrument

The following story comes from a single 8-minute video I shot as I interacted with a pair of boys near the end of their very first session with MelodyMorph. In this session, we had introduced the app using the scavenger hunt concept, in which students called out different features of the app as they discovered them, and then we discussed as a group. The remainder of the class time was used for open-ended exploration. Henry and Jacob rapidly developed a sequence of ideas culminating in a musical loop (a "loop" here in its more common musical sense of a repeating pattern) with its own instrument for accompaniment.

Henry and Jacob's first three experiments are shown in the figure below. First, Jacob pulled ten vibraphone bells onto the canvas, and created a wandering path playing through them. Next, he created a more deliberately shaped symmetrical construction of pluck bells, and created a path through them that connects back to its beginning, making an infinite loop. Then Henry took over with a new idea. Having heard the looping pattern, he decided to create his own loop. He pulled out a combination of electric piano bells with high and low pitches, and paired them vibraphone bells. He struggled a bit with the path-making function of the app, because he was trying to record a pattern of rhythmic taps. He explained that he was trying to "make a beat," meaning a repeating percussive pattern. The paths can only record continuous sliding movements, not separate taps, so he had to try to find another approach. After a bit more exploration, he discovered that the path player could make very short loops, resulting in rapidly repeating notes. Unfortunately, connecting two of these fast loops together triggered a bug in the app, causing it to crash. My explanation and apology for this bug led us into a brief conversation about how I had made the app, and Henry was very interested in learning about the code. I tried to bring our focus back to the musical aspects as we restarted the app.
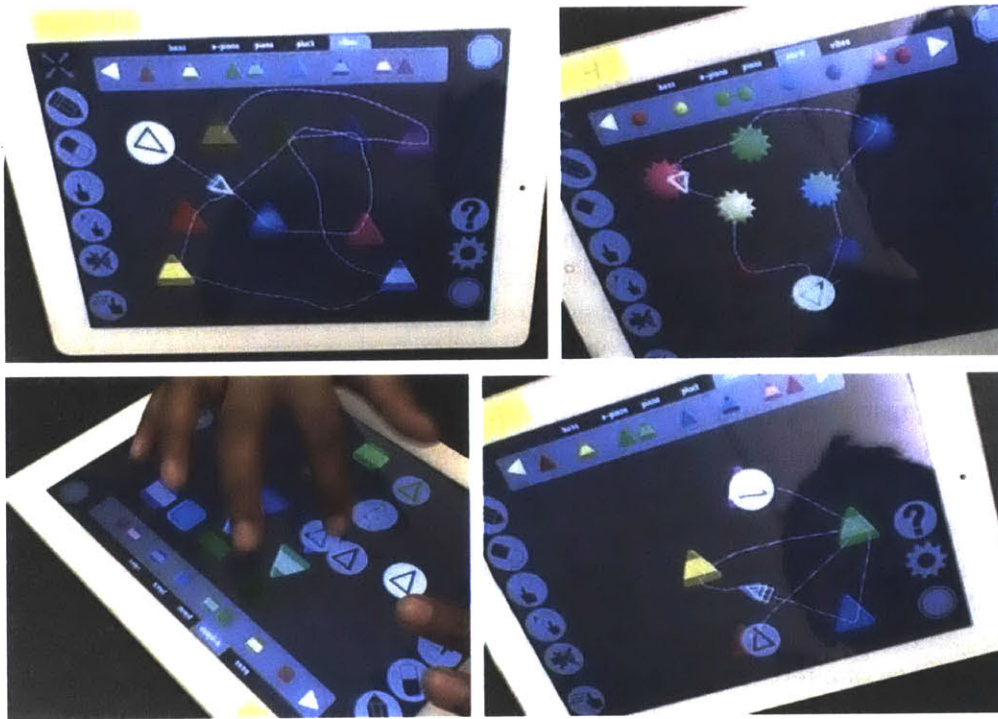
Figure 23: Three path experiments. Top left: a wandering path through vibraphone bells. Top right: a loop through pluck bells. Bottom left: an attempt to "make a beat." Bottom right: a pentagonal path.

Next, Jacob took over, first making a short path through a pentagon of vibraphone bells (figure 23, bottom right). He deleted this first pentagon, and made a new one, connecting the path into a continuous loop (notated at bottom of figure 24). The rhythm of my notation is only approximate, but shows the loop as a pattern in 5/4 time. This odd meter may have emerged spontaneously or by chance from the process of tracing the five-note pattern. It seems unlikely that Jacob intentionally or knowingly performed the pattern in 5/4 time.

With this stable and interesting pattern continuously looping, Jacob began to add additional bells. He used these as an instrument for improvising melodic patterns along with the loop. First, he added a vibraphone bell F to the center of the pentagon, and two pluck bells, a G and an A (top right of the figure). He played a repeating pattern on the two pluck bells, and I asked him what he was doing. He replied "I don't know," to which both Henry and I replied with encouragement, saying we thought it was a good idea. Next Jacob removed these accompaniment notes, and replaced them with four new ones (bottom left of the figure), C, D, A and C'. After playing some additional repeating patterns on these, he replaced them again with four piano notes, D, E, B, and C', with again an F in the center of the pattern. Again, he played some melodies over the looping accompaniment.
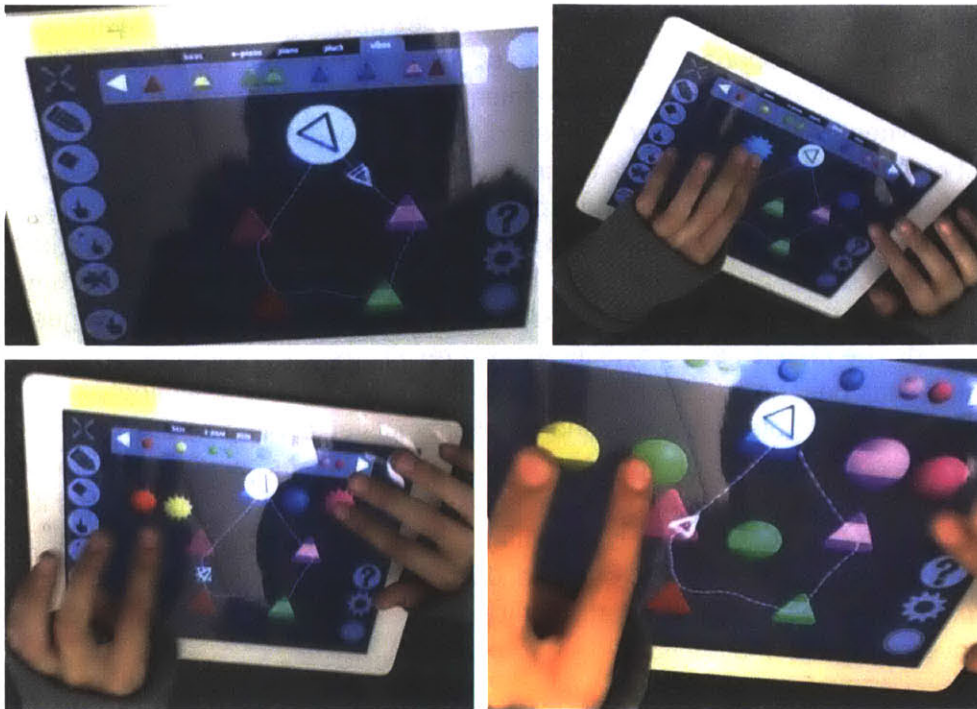
Figure 24: A pentagonal loop with accompaniment. Top left: a looping path through five bells, triggering pitches B F C C' A. The same loop is used throughout these screenshots. Top right: the loop with accompaniment played on two pluck bells, E and A, and a vibraphone bell at the center, F. Bottom left: the loop with with four pluck bells, C, D, A and C'. Bottom right: the loop with piano bells D, E, B and C', and at the center, F. Below: the pentagonal loop notated. Note that notated pattern starts on the C', at the upper left of the pentagon.

Notes: Complexity in simplicity

Watching the video of this episode, and then messing around with my own reconstructed version of the five-note looping pattern, I noticed a fascinating ambiguity. What is the musical key of this loop (notated in figure 24)? What pitch does it want to resolve to? It contains both a low and a high C, and the notes are all drawn from a C major scale, so this was my initial assumption. But after playing around with it, my impression is that the loop is basically in D dorian minor, moving between that chord and an implied "four chord," the G7. But this conclusion is made uncertain by the fact that neither a D nor a G appears in the loop.

In this episode we see an exploration of several different types of spatial map. We start with Jacob randomly pulling out vibraphone bells onto the canvas and making a wandering path through them. Then, he arranges pluck bells, more intentionally this time, in a symmetrical pattern with a loop. When Henry takes over, he tries to make the looping music fit better into a musical idea that he imagines, a "beat," but a technical limitation prevents him from trying out this idea. Next, Jacob simplifies further, making a pentagon with a path crossing within it, and then a pentagonal loop with the path around its edge. In this process, we see the intentionality in the design process increase, as they choose a smaller number of bells, in more geometrical shapes, with simpler looping paths. These changes in the tinkering map reflect the changes in their musical thinking. The map is getting simpler, but the result has a new musical complexity. The loop has both an odd 5/4 meter and an ambiguous tonality, seemingly implying but omitting the root note. Jacob's sequence of attempts at creating a satisfying instrument to play along with the repeating pattern could be seen as musical inquiry. Perhaps he was searching for the root note, without being able to articulate this goal explicitly. And it is possible that Jacob arrived at the same conclusion that I did. In his final iteration, he has a D bell at far left, suggesting the use of D as a root note. Jacob's series of attempts to understand the pattern's properties form a particularly explicit example of a tinkering loop made up of a conversation with materials.

Loops and Maps in the Vignettes

These three vignettes provide glimpses into the explorations of other students in our MelodyMorph classes: a musical marriage proposal, a morning melody, and the evolution of a looping instrument. In each of them, we see evidence of the loops and maps in musical tinkering processes.

In the marriage proposal, the students communicate emotions like nervousness and excitement using a musical map. Their pair of melodies is generated by a transformation, preserving the melodic contour of the first melody, but shifting the pitches to differentiate the characters in the story. This visible transformation of their map helps to show their musical thinking.

In the morning melody, the student expresses the mix of emotions about waking up in the form of a melody that changes as it repeats, its ending shifting from resolved to unresolved. In this case, the six notes on the screen function as a tinkering map that let him tell me a story about his musical thinking, when he "mixed it up" so that the first and second sets of notes were not identical. Interestingly, this simple change does not explain how he thought about the more complex melody he recorded using those same notes.

The evolution of the looping instrument shows us a tinkering loop in the form of an iterative design process, with constructions that grow simpler on the surface, but reveal increasing musical complexity. And the changing shapes of the loops are tinkering maps that reflect the students' changing musical thinking.

Reflections

Here I review my observations across the case studies of MelodyMorph in action, addressing some of the questions I raised in the introduction about loops and maps in musical tinkering. I look at musical backtalk, reflection-on-action, and collaborative emergence. Then I explore the distinctions between figural and formal, and between map-making and path-making. I arrive at some new concepts, which I call landscape-making, road-making, and melodic navigation. I look at the connections between musical inquiry, linking, and strange loops. And finally I show how tinkering maps and tinkering loops are intertwined.

Musical Backtalk

Backtalk, in Schön's sense, is surprising feedback in a conversation with materials. What kind of backtalk appears in musical tinkering loops? In Sandy's very first experiment, he creates a simple row of five bells, and discovers in it, apparently by chance, a familiar melody. The backtalk in this kind of situation is an interaction, between a random sequence of notes played on a set of bells, and our musical minds' proclivity for pattern recognition. We seem to easily form associations between random scraps of melody and even very dimly remembered tunes. I have observed this phenomenon, a kind of Proustian Madeleine effect in music, in myself and occasionally in others as well. But unlike the usual musical remembrance, which is fleeting, when it happens in MelodyMorph we can pursue it. With MelodyMorph, the melodic material stays on the screen, so we can play with it, reconstructing and even reimagining a remembered melody.

Musical backtalk can also be triggered through movement and gesture. Sandy's paths with different shapes lead to different "catchy" melodies using the same bells, but his attention is not focused on composing a melody or a rhythm. Instead, he is experimenting with shapes and symmetry, and with the gestural movements that create the path. These gestures interact with the bells already on the screen to yield new musical phenomena. The backtalk from these experiments raises questions about how these shapes and movements relate to melodies. What will it sound like in reverse? Or sideways? Sandy's musical inquiry process is stimulated by musical backtalk.

Reflection-on-Action Without Words

What is the role of reflection-on-action in musical tinkering loops? These are situations in which a learner steps outside of a situation in order to analyze it. It seems difficult to find direct evidence of reflection-on-action in these case studies. One place to look is in things students say aloud, describing what they have been working on. In my coding scheme, I refer to these moments as "formalizing," specifically for expressing ideas in words. But the explicit expression of musical ideas in words was fairly rare in my case studies, and throughout our classroom sessions. Sandy does formalize occasionally as he works out musical puzzles. He uses a metaphor for the melodic contour notation, and describes his loops and canons. He says that two of the melodies in the cartoon should start at the same time. He works out the structure of the layered melodies in the My Singing Monsters game. Ben's answers to my

questions about how he composed his morning melody also include some formal expressions of musical relationships, especially the idea that different musical properties such as pitch and loudness can be used to express different emotions.

Perhaps musical tinkering is difficult to reflect about in words. When we asked the students questions as they worked, they struggled to answer them verbally. Instead they often simply pointed to something on the screen, or triggered a recorded melody. Musical phenomena are inherently difficult to describe, because musical experiences are continuous and move through time. This makes them difficult to point to, in order to refer to them. Music notation, or an arrangement of bells on the MelodyMorph screen, can partially capture these phenomena and make them static. Without the musical vocabulary, the students' reflection-on-action may have been expressed more in their musical material on the screen than in words.

Visible Collaborative Emergence

We see a complex collaborative tinkering loop as Sandy and Daniel create a musical cartoon. Sawyer (2002) describes a synergy between two kinds of feedback in an improvised theater performance, that may also be present here. In "collaborative emergence," the creative work assembles itself from many individual moves by the group. In "downward causality," the properties of the emerging work constrain which successive moves are possible, accelerating the emerging coherence of the piece.

The melodies that Sandy and Daniel composed for the smiley guy and the spiky guy may have emerged from this kind of feedback process. The two melodies are clearly related, being in some ways similar, and in other ways opposite. Both consist of two phrases played on one row of bells, traversing the row in one direction and then the other. But one traverses up then down, and the other down then up. And one is minor, the other major. Traces of this pattern appear in the later melodies in their cartoon as well. The complex inter-relationships in this set of melodies did not arise from a top-down plan. Instead, the melodies were built through a sequence of intuitive moves, that gradually caused them to cohere. As a pattern emerged in one melody, it may have provided a constraint on the others.

Sawyer's scenarios occur in improvised theater and music, in which these emergent phenomena flow through time. They leave no visible trace. The properties of MelodyMorph as a collaborative composition tool may have made collaborative emergence more visible. The elements of a MelodyMorph collaboration are not just momentary sounds, as in a theater or jazz improvisation. The elements are also graphical objects on the screen, that persist over time. It may be easier to build on a collaborator's moves when the results are visible, and hold still to be understood, imitated, and manipulated.

Maps More Figural than Formal

Do we see a distinction in MelodyMorph similar to the one that Bamberger (2013) observes in children's invented notation, between "figural" and "formal" drawings? Figural drawings reflect

intuitive perceptual groupings, while formal drawings use a more systematic or quantitative notation. Are the tinkering maps in MelodyMorph figural, formal, or something else?

Generally, the placement of bells in the students' creations did not reflect a formal measurement of pitch or time. Instead the bells seemed situated to make them easy to play, for example in a row that one traverses forward and back to play a melody. In other cases the spatial arrangement may have reflected a rhythmic pattern, but not in a metric way. This occurs in Sandy's two-note groups that became two note rhythms (figure 5) and in the five-note ring of the looping instrument (figure 24).

Spatial arrangement can also be used to represent pitch, as in a melodic contour where height corresponds to pitch (as in Blair's (2007) work with "music maps"). This mapping could work in a formal way, as a precisely measured contour determined by the pitches in the melody. Or it could work in a figural way in MelodyMorph, as an approximate melodic contour showing the overall shape of the melody. There are hints of this figural style, but no clear evidence that the students used either one. In Sandy's smiley guy melody, for example (figure 8), the ascending pitches are positioned on an ascending diagonal arc. Daniel's melody ascends on a similar diagonal (figure 9), but his melody also includes a descending interval, which is not reflected in the shape of the row.

It seems that the students' tinkering maps in MelodyMorph did not use space to represent either time or pitch in a formal way. We do see a kind of figural notation in their use of separate groupings for separate musical phrases. But mainly it appears that students used space on the screen to adapt their arrangements to their hand movements as they played. We see patterns of bells, and paths through them, forming arcs, rows, and rings. In the case of the looper instrument, the added accompaniment bells appear on both sides, in a way that makes it easy to play them with two hands. All these patterns reflect the shapes of the hand movements used to create and play them. In some cases perhaps the musical structures came first, and inspired the shapes. But it may be that the shapes originated in intuitive gestural movements.

This observation, that gestural movements shape the use of space in MelodyMorph, raises another question: will we see the strategies that Bamberger calls map-making and path-making?

From Map-Making to Path-Making and Back

In Bamberger's observations of children constructing tunes with physical bells, she saw two distinct strategies. Path-makers put the bells in a row in the order of the melody, in order to follow an "action-path" of movement along the row. Map-makers, by contrast, first build a scale, with the pitches in order from low to high. Having established this fixed reference structure, they can then work out how to play the melody by moving around on it.

One question we might ask about MelodyMorph is whether we will see this same distinction among students. The students do indeed sometimes construct scales, and other times construct paths across rows of bells. But the situation is different in most MelodyMorph activities from the ones Bamberger worked with, because in most cases the students are not attempting to reconstruct a known melody. Instead, they are working on their own creative compositions, which may require a different set of strategies.

First I will look at tune reconstruction in MelodyMorph, the situation that is most similar to the one in which Bamberger describes path-making and map-making. Do we see the distinction there? Bamberger (2013) also writes about a transition between the two strategies. She describes a student named Conan, who moves bells around as he tries to play the melody to "Twinkle, Twinkle, Little Star." As Conan assembles the tune, he lines the bells up in a row. By the time he is finished, the row is in the order of a musical scale. He did not construct the scale at the beginning, as a map on which to navigate the tune. But once has completed the tune, he is able to use the positions of the bells as numbers in a notation that would let another person play the tune on his row. Through tinkering with paths, he creates a map, in this case a fixed reference space for pitch. Will we see this same transition in MelodyMorph, from path-making to map-making? Bamberger acknowledges that map-making is not intrinsically superior to path-making. She suggests instead that the goal might be to "have access to both action and symbol such that one is able to choose depending when, where, and what for."

When Sandy is working on reconstructing the Furcorn melody, we see a moment when he actually seems to shift from map-making to path-making. This is the opposite direction from Brad's shift. As Sandy puzzles out the second part of the Furcorn melody (figure 18, and following text), he begins by trying to play it using a descending scale he has constructed. But after some struggle, he rearranges the bells to partially reflect the ordering of the melody, which has intervals that move down, then up, then down again. He may have lost confidence in his use of the scale, not having a strong grasp on the remembered melody he was trying to reconstruct. So, I speculate that he decides to switch strategies to make it easier to play. The path for the melody on this new arrangement is simpler. With the scale-ordered bells, playing the melody requires skipping over a bell, and changing directions twice. But the path through the melody-ordered arrangement is always between adjacent bells, and only changes directions once.

We also see this transition at the very beginning of Sandy's story, in which he starts with map-making. He constructs a scale (figure 1), and then uses it to discover a familiar melody, and experiment with paths. This is a different situation from Bamberger's tasks, because he is exploring without the goal of reconstructing a particular tune. In Bamberger's case, the tune comes first, and creates the need for a scale. In Sandy's case, the scale came first, and yielded up the melody through play. Making this scale is a different type of map-making than the strategy Bamberger describes. Sandy was not using the map to trace a path to a destination. Instead, he was constructing the map as a musical landscape to explore.

Landscape-Making and Melodic Navigation

We need a different set of concepts from Bamberger's maps and paths to describe the ways in which the students tinker with MelodyMorph. I will use the term "landscape-making" to describe the creation of an arrangement of bells that is meant to be explored. It is a landscape because it is a space of musical possibilities, a connected set of musical places. There is no predetermined pathway through it. A musical landscape is a kind of tinkering map, because its layout reflects a musical thought process, and this layout can be fluidly changed to reflect new ideas. It also has the property of tinkering maps I call "linking," because it is simultaneously a map of a musical world, and a way to explore that world in itself. Interacting with a street map on a computer screen lets you explore a city, but it does not actually cause you to move around in it. These musical landscapes both symbolize musical ideas, and allow you to explore and experience them directly.

Playing music by moving through this landscape thus becomes "melodic navigation." This is playful, spontaneous navigation, without a predetermined goal. It is what Suchman (1987) calls "situated action," which is a way of acting by continuously responding to the ever-shifting context around you. Melodic navigation is a continuous process of making musical moves, listening to the results, and responding by making new moves into the nearby landscape. Improvisation on a traditional musical instrument can also be a kind of melodic navigation, because each instrument is itself a landscape of possibilities. But tinkering maps allow for a special kind of melodic navigation, because you can craft and transform your own landscape as you navigate.

Ben's morning melody (figure 22) is an example of a musical landscape. It has a seemingly simple structure, which is a row of six bells. But Ben plays a melody that expresses how he feels in the morning by using melodic navigation. He does not simply play the bells one after another from left to right. Instead he plays sequences of adjacent bells, sometimes reversing direction, and sometimes skipping bells. His path is not prescribed by the landscape, but it is not arbitrary, either. He makes use of a logic embedded in the sequence of the bells to generate a melody. The internal coherence of his melody results from the fact that he is moving across a fixed pattern, with its own internal relationships. The different parts of his melody are related to each other because they are made using different traversals of the same stable ordering.

We also see melodic navigation in Daniel's construction of his spiky guy melody (figure 9). He works on his composition by navigating through his row of bells, moving back and forth as he looks for an ending. Deciding that something is missing, he navigates a bit in the palette, and finds a new bell to add to his landscape. Finally, he tries again, making a new melodic path. He has modified his landscape in order to navigate it in a new way, and reach a new goal.

Melodic navigation may sometimes be guided by physical gestures as much as by listening. Sandy makes some seemingly arbitrary arrangements of bells and uses these to compose
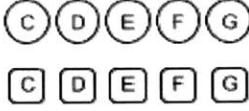
melodies, as in figure 5. In this case, he uses the feature of MelodyMorph that allows him to trace a path on the screen that he can then replay. His drawing gestures alternate up and down in a symmetrical pattern. These hand movements seem to determine the melody, rather than the melody determining his movements. He listens critically to the results (deciding the melody is "catchy"), and tries out alternate paths, which can be seen as experiments in melodic navigation.

Road-making

In the marriage proposal melody (figure 21), the bells are ordered in a strict correspondence with the melody. The melody is played by moving across the row left to right, playing the bells each once, one at a time. As a compositional process, was this an example of landscape-making? This special case, constrained to a straight line, we might instead call "road-making." It has something like the "formal" quality discussed above in contrast with figural notations, but not because it measures out time or pitch. The formality here is in preserving a strictly sequential ordering of the melody.

The Many Maps of MelodyMorph

Between figural and formal structures, map-making and path-making, landscape-making and road-making, there were a variety of ways in the students used space to represent musical ideas in MelodyMorph. The table below shows twelve types.

Musical Property	Spatial Arrangement	
Scale order	Row, left to right	
Melody order	Row, left to right	
Figural phrase boundaries	Horizontal groupings	
Multiple phrases	Multiple rows	
Multiple timbres	Multiple rows	
Ascending pitches	Diagonal row	

Fit under fingers	Arc	(D)(E)(F)(G)(C)
Split between two hands	Clusters on left and right	(E)(A)(D)(G)(C)(F)
Figural rhythmic groupings	Short columns	(E)(A)(D)(G)(F)(B)(E)(A)
Wandering melody	Arbitrary pattern	(E)(D)(G)(C)(F)
Spatial rhythmic pattern	Geometric figure	(E)(D)(G)(F)(C)
Repeating pattern	Ring	(A)(E)(D)(G)(F)(C)

Table 2: Twelve ways in which students mapped spatial arrangements of bells in MelodyMorph to musical properties.

There were also a number of possible map types that did not appear in the students' work. These are arrangements that I have played with myself, or observed others using. These arrangements are more often made by people with a musical background of some kind.

Musical Property	**Spatial Arrangement**	
Figural rhythmic pattern	Horizontal spacing	O O OOO O O OOO
Formal rhythmic pattern	Horizontal spacing	O O OOO O O OOO
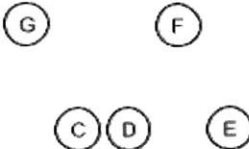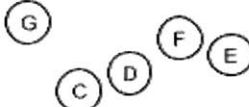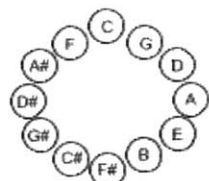
Figural pitch height	Varying height along row	
Formal pitch height	Varying height along row	
Interval relationships	Grid pattern	
Harmonic relationships	Radial pattern	

Table 3: Six possibilities not used by the students for mapping spatial arrangements to musical properties.

These tables show that MelodyMorph supports a variety of different approaches to using space for mapping musical properties. Supporting this kind of diversity is important in tools for making tinkering maps. If the maps are going to be able to reflect diverse processes of musical thinking, the tool needs to support diverse uses of space.

What can we learn from the set of possible mappings that were *not* used by the students? These include formal mappings for pitch and for rhythm. These two strategies are ubiquitous across music contexts, and are commonly accepted as the most useful ways to show musical parameters using space. Typical computer music interfaces work like graphs of pitch over time, with time mapped to the horizontal axis and pitch to the vertical axis. Standard music notation uses a related mapping on the two axes. But the students did not create any similar measured mappings of space to time or to pitch in MelodyMorph. Perhaps this is because the use of the app did not create any necessity for these formal mappings.

Standard notation needs to use a quantitative symbol system for rhythm (ratios between durations), in order to allow the repeatable and precise recreation of rhythms. Formal systems in invented music notation similarly aim for this precise repeatability. But MelodyMorph arrangements do not have this requirement. Accurate and repeatable rhythmic timings can be recorded in the app, using a recorder bell to record a sequence of taps, or a path player to record a set of sliding movements. Time in MelodyMorph is embedded in recorded actions of

96

the user, not in spatial relationships. The rhythm of a melody is performed as a series of taps, or a set of sliding moves recorded in a path-player. Because the app does not automate playback using spatial arrangements to control time (as in for example a step sequencer), the spatial arrangement has no functional role for this mapping.

Similarly, pitch in MelodyMorph is embedded in the bells themselves. It is visible as a color and optionally a letter name for the pitch class. These properties are stable, not varying as you move the bells around. Any additional system for mapping pitches, such as their position on an axis, is redundant.

One might imagine that, even though these formal mappings are not as useful in the context of the app, the figural versions would still be useful for conveying musical ideas. We do see hints of figural relationships. They are just not the same as the ones we see in hand-drawn invented notation. Instead of approximate melodic contours, we see simpler arcs and diagonals. Instead of figural rhythmic groupings, we see phrases more coarsely separated, in horizontal groupings or in rows.

The figural qualities in MelodyMorph arrangements that reflect musical ideas are driven more by functional uses than notational needs. The arrangement of bells in the order of a melody or of a scale reflect the approach the student used to compose and to play the melody. Similarly, the structure of musical landscapes reflects their approach to musical navigation.

As I mentioned, the approaches in table 3 were more often used by people with a musical background. Why did they use them? Perhaps these strategies simplify their creation process, by organizing the space using mappings already familiar from other systems. The grid and radial organizations are also ways to organize pitch and harmony that these users probably derive from existing diagrams, as a way to help structure their musical inquiry processes.

Linking and Musical Inquiry

MelodyMorph has the property of tinkering maps I call linking, which is the ability to both symbolize musical structures, and cause them to play. We see this property in computer programs, and in magic spells, which are made up of words that both symbolize actions and cause them. MelodyMorph creations are made up of spatial arrangements on the screen that symbolize musical structures and can also be used to play the music. How does this special property change how we interact with and compose music?

Throughout these case studies, we see examples of musical inquiry processes. The students are continuously asking and answering their own questions. These inquiry cycles are tinkering loops that move across all the activities of experimenting: anchoring, exploring, iterating, and transforming. They also crucially involve the activities of expression: mapping and formalizing, storytelling and emoting. The inquiry process even moves through the social realm of connecting and collaborating. The linking property of MelodyMorph creations is what connects these different levels of the inquiry process together. To experiment, the students need to be

able to play with their ideas, manipulating and transforming them as symbols. But they also use their creations expressively, performing and sharing them. The linking property of MelodyMorph joins these activities together, in one system they can use to both manipulate symbols and make sounds. Perhaps this complex tinkering loop is what Hofstadter (1979) calls a "strange loop."

A Strange Tinkering Loop

A strange loop is a feedback loop that crosses informational levels, with surprising results. Another way to describe it is that something inside of the system seems to pop up and act on the system, as though outside of it. Language does this when it talks about itself, which can lead to surprising conundrums, such as the sentence "this sentence is false." Hofstadter writes about strange loops arising from "tangled hierarchies." As we move up or down through a tangled hierarchy, we unexpectedly find ourselves back where we started. The linking property of MelodyMorph compositions may create such a strange loop by conjoining symbol and action, which are at different levels of a hierarchy. In the backtalk between a tinkerer and her musical material, for example, we see a melody move across levels. It goes from a visual representation to an unexpected musical insight and back, through a gesture on the screen. In this moment we see a tinkering loop and a tinkering map interact.

Loops and Maps Intertwine

The more closely I look at the phenomena of tinkering loops and tinkering maps, the more I see their causes and effects intertwined. Tinkering loops across time scales rely on the properties of the tinkering map. Tinkering maps in turn are formed through iterative feedback loops.

Reflection-on-action in MelodyMorph may rely on the ability of tinkering maps to externalize musical ideas, making them visible. Verbal reflection, at least for the students I worked with, was not as effective as reflecting through interaction with the maps. And the feedback loop of collaborative emergence is amplified by this same visibility of ideas in tinkering maps.

The structure of the tinkering maps created by the students seems partly driven by the inner-loop feedback process of embodied musical thinking, which we observe as hand gestures. These hand movements follow melody-ordered or scale-ordered bells, trace paths to make looping melodies, and navigate melodic landscapes. Gestures can directly shape tinkering maps.

Some figural notation-like arrangements appear in the tinkering maps, but few formal ones. In spite of the lack of notation-like features, we see a diverse set of spatial arrangements in the students' work. These diverse approaches reflect the diversity in the students' musical thinking, which develops through the context-loops of social feedback.

Understanding this intertwining of loops and maps leads to a new way of seeing the creative potential of MelodyMorph. The combination of tinkering loops and tinkering maps creates a

new musical medium. It closely couples together instrument-making, composing, and improvisation. The creative activities of landscape-making and melodic navigation are part of the new vocabulary of this medium. Perhaps this fertile combination of tinkering loops and tinkering maps, yielding up strange loops and landscapes, will help us design more powerful systems for musical tinkering.

Chapter 4:
Designing MaKey MaKey

In this chapter I present the design of the MaKey MaKey invention kit, focusing on its use for musical tinkering. I present some initial inspiration for the design from previous projects, and then introduce the kit itself, and some concepts that guided its design. I discuss some related projects, and conclude with ideas for future work.

Inspiration

Here I describe some prior projects that provided inspiration for the MaKey MaKey, including the Cricket kit, the Scratch sensor board, Drawdio, and keyboard hacking.

Cricket instruments

My exploration of musical tinkering began when I joined a workshop using the Cricket family of programmable electronics kits. The PicoCricket robotics kit (Rusk et al., 2008) is designed for open-ended tinkering with a variety of inputs, such as light, loudness, touch, and electrical resistance, and outputs including motors, colored lights, and musical instrument sounds. The PicoCricket itself is a "programmable brick," a small computer that can run a program created by the user on a PC and then downloaded to the brick. Over the course of several years I experimented with the PicoCricket's predecessor, the Cricket (Resnick et al., 1998), and later the PicoCricket, creating a variety of musical contraptions. These involved writing a simple program in the Cricket Logo or LogoBlocks programming language, which transformed the input from a sensor into musical output, played by an attached MIDI synthesizer. As experiments in the design of physical-digital musical instruments using everyday materials, they were conceptual precursors to the design of MaKey MaKey.
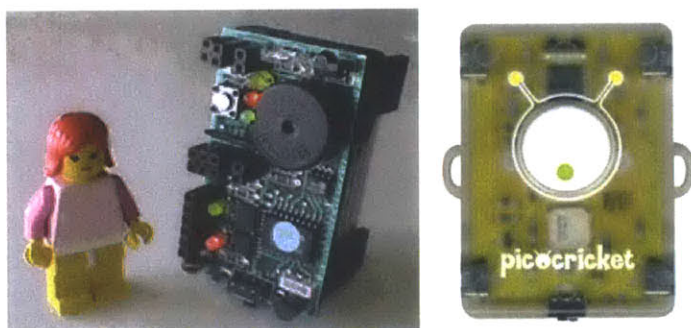
Figure 1: Programmable bricks. Left: the "blue dot" cricket, with a LEGO mini-fig. Right: the PicoCricket.

The "Wandering Wands" was a Cricket instrument that used two light sensors, each attached to the tip of a decorated plastic tube. The user would wave the two wands around, one

controlling pitch in a chromatic scale and the other selecting an instrument timbre using the light level. The wands rewarded exploration of light and shadows in different parts of a room with a variety of sounds. A later modification that I made to the program for this instrument turned it into the "Monkifier." Rather than simply mapping the integer light sensor values to chromatic MIDI notes, I transformed the values so that they were all odd numbers. Musically, this results in a "whole tone" scale, which has an ambiguous unresolved sound as you play notes up and down the scale. The scale's name is a reference to the jazz pianist Thelonious Monk, who often improvised fast melodic lines running up and down the whole tone scale, and used it in his compositions.
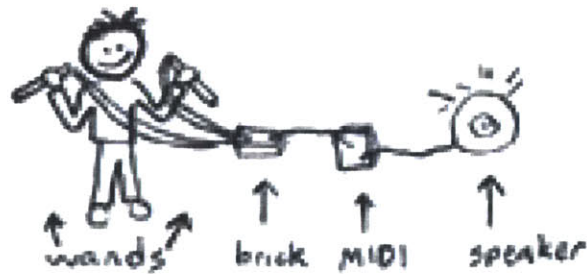
Figure 2: Wandering wands. Left: the wands in action, with the cricket and speakers in a bag. Right: My diagram of the system.

Several of my Cricket and PicoCricket instruments used resistance sensing as the input to control the pitches of musical notes. These included the Ice Theremin and Electroloops. The "Ice Theremin," inspired by a project from our collaborators at the Exploratorium's Tinkering Studio, consisted of a ball of ice that users could touch to trigger sounds. The user would connect their body to a ground connection, then touch the ice, completing a circuit. The resistance in this circuit controlled the pitch of the sound in a chromatic scale. The resistance would vary due to the conductivity of the skin and of the ice, and the amount of contact between them, resulting in an expressive and magical-feeling, if slightly unpredictable, instrument.
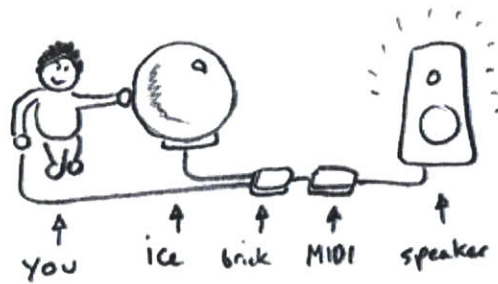
Figure 4: The "Ice Theremin." Left: a child plays with a version of the instrument, completing a circuit to generate sound by touching a foil-tipped wand to a ball of ice. Right: my diagram showing the parts of this resistance sensing instrument, including the Cricket programmable brick.

"Electroloops" was an instrument for two players, controlled by each player using a pair of rods. The rods were tipped with conductive foil. The user held a rod in each hand, squeezing them to complete a circuit, controlling the pitch using a resistance measurement. One pair of rods controlled an electric piano sound, the other a bass sound. The sounds were quantized into a steady pattern of short notes, and continuously overdubbed into a loop. Each squeeze of the rods added new notes into the looping melody.

Figure 4: Electroloops, a music looping instrument for two players.

Scratch Sensor Board instruments

The Scratch Sensor Board (Millner, 2008), later known as the PicoBoard, provided another flexible platform for inventing musical instruments that mapped sensor input to pitch. I documented several of these as I created sample projects to include with a version of the Scratch programming environment (Resnick et al, 2009). The Scratch Sensor Board is able to measure the resistance in a circuit and send it to a Scratch project on a computer which can play musical notes, in a way similar to the Cricket and PicoCricket. The "Fruitophone" used a row of small chunks of melon as a variable resistor. The user could trigger different notes by holding an alligator clip on a melon chunk at one end, and touching the other clip to different melons along the row. We also experimented with making similar musical instruments using

graphite pencil drawings on paper, wet string, plants, and chains of people. In many ways, these instruments are similar to those I made using the Cricket, mapping resistance sensor input to musical pitch using a simple program. One important difference, later reflected in the design of MaKey MaKey, was that the program ran on a computer. Unlike the smaller and more limited programmable brick, the computer could be used to do much more than playing the pitches. It could for example display information on the screen, such as an animation or a visualization of the sensor readings or the musical output, or provide additional controls for adjusting the mapping and the sounds.
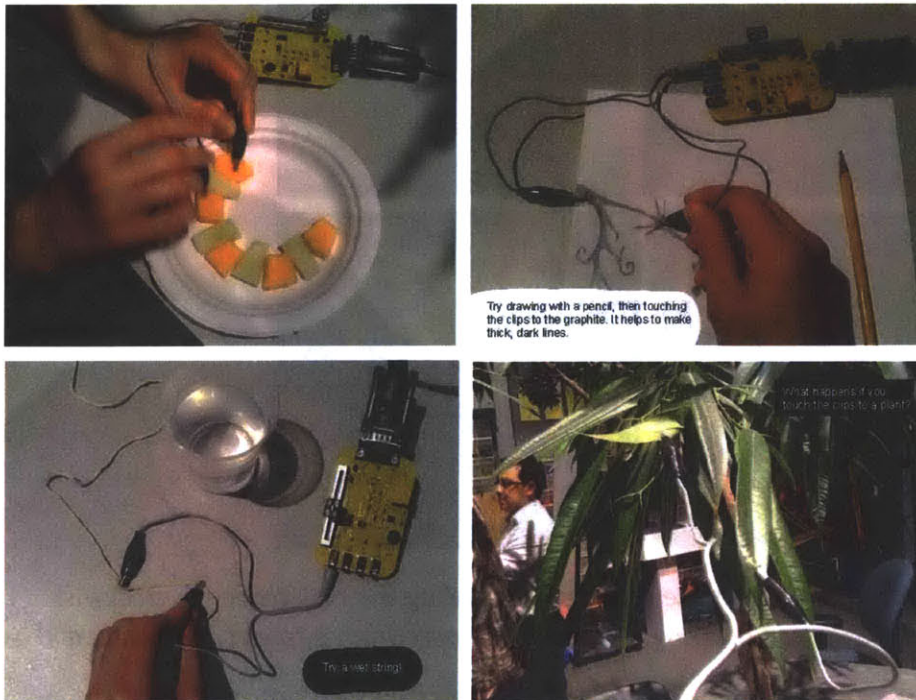
Figure 5: Scratch Sensor Board musical instruments. Top left: a melon piano. Top right: a musical pencil drawing. Bottom left: A wet string instrument. Bottom right: musical leaves.

Drawdio

The Drawdio invention kit, created by Jay Silver (2009), also served as a precursor to MaKey MaKey. Like the Cricket and Scratch Sensor Board instruments, it measures electrical resistance and uses it to control the pitch of a sound. Unlike the other systems, it is limited to this specific relationship between input and output, and the relationship is continuous, or analog: a smoothly changing resistance measurement controls a smoothly changing pitched sound, rather than playing discrete notes in a scale. This continuous sensing makes it possible to control the sound in a very direct and expressive way. But this same continuous control makes it very difficult to use the Drawdio to play melodies. The difficulty is analogous to the challenge of learning to play a violin, as opposed to a guitar which has frets showing

the placement of individual notes. Drawdio can also sense a much greater range of resistances than the other systems, up to tens of MegOhms. This wide range opens up a greater range of possibilities for connecting everyday objects and natural materials to the kit to make musical instruments.

Figure 6: Drawdio inventions that make music by measuring the resistance of, clockwise from top left: a graphite pencil drawing, a stream of water from a faucet, a flower, and a watercolor painting.

Keyboard hacking

Another important precursor to the MaKey MaKey, in addition to our experiments with the Cricket, ScratchBoard, and Drawdio, was the practice of "keyboard hacking." The need for keyboard hacking arises when you want to create a new physical input to a computer. Rather than create a custom device that can send data to custom software on a computer, it is often easier instead to build something that makes use of the existing system by which keyboards send keypress events to the computer. One way to do this is to "hack" an existing computer keyboard, breaking it open and connecting new buttons to it. Suppose, for example, that you are making an interactive kiosk that requires a single large button. One way to do this is to connect wires to the circuit board inside a keyboard at the location of the "space" key, and then attach those wires to an arcade button. Now, pressing the arcade button has the same effect as pressing the space key. I made a few interactive installations in this way, to allow people to interact with software I had created. One of these is my playful music looper, "MmmTsss" (Silver and Rosenbaum, 2010), which already had simple keyboard controls built

in. In one case, I used a similar process to keyboard hacking but with a computer mouse, in order to run wires down to a foot pedal that could click the mouse. In another case, I performed an even simpler keyboard hack by gluing laser-cut wooden buttons to several keys, and hiding the rest of the keyboard with fabric.

Figure 7: MmmTsss looper kiosk setup. The music looping software is controlled by laser-cut wooden buttons connected to a keyboard and mouse which are hidden by fabric.

The MaKey MaKey Invention Kit

The MaKey MaKey project, a collaboration with Jay Silver, began with the question of how we could make a construction kit that was even more open-ended and even more accessible and easy to use than the projects that had come before. Our central idea was to combine open-ended creation in the physical world and in the digital world. We knew from our resistance sensing experiments with Drawdio that connecting everyday objects to that circuit opened up a large space of creative possibilities. But Drawdio was limited to generating a buzzing tone. How could we connect this very flexible input to an equally flexible output? The breakthrough came from our experiences with keyboard hacking. We decided to make a device that could connect to a computer and generate keypresses. Now, any software normally controlled by a keyboard could be controlled instead by everyday objects connected to the circuit.

The MaKey MaKey is a small circuit board that connects via USB to a computer. Using the HID protocol, it emulates a standard computer keyboard and mouse, so that it can control the computer by sending keyboard and mouse events. The kit comes with a set of alligator clips and wires which allow the user to connect everyday objects to the circuit, and turn them into keys. There are several input connections on the circuit board, labeled with the keys they control. For example, you can connect one end of an alligator clip cable to the "up arrow" on

the MaKey MaKey, and the other end to a banana. If you connect your body to "ground" or "earth" on the MaKey MaKey, you can then touch the banana to complete the circuit. When the MaKey MaKey detects that the circuit is completed, it tells the computer that the up arrow key has been pressed.
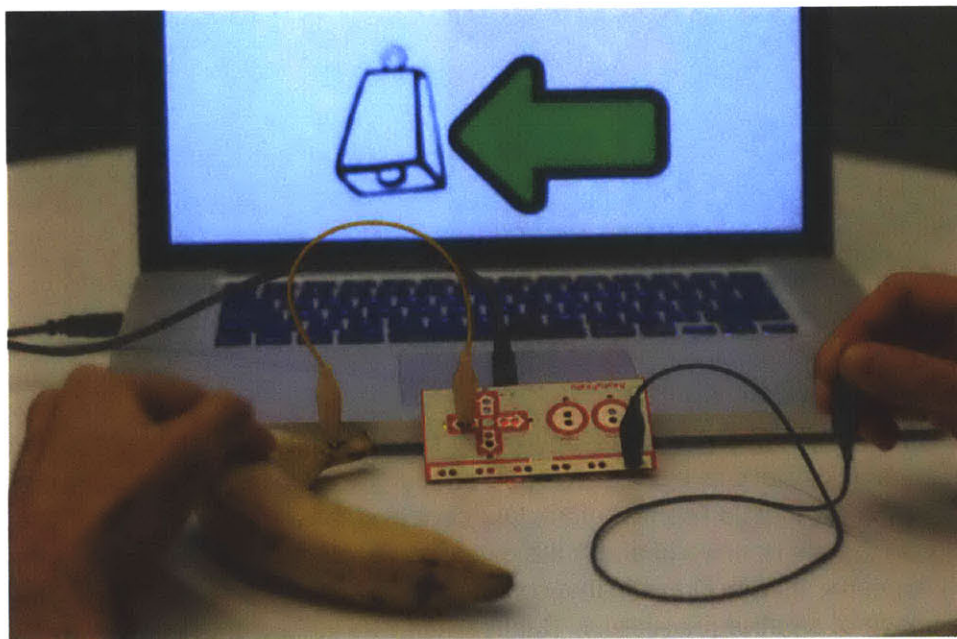
Figure 8: The MaKey MaKey kit. Top: the MaKey MaKey detects when a circuit is completed, and makes the computer think a key has been pressed. Bottom: completing a circuit by touching a banana, to trigger a cowbell sound.

Initial prototypes

I assembled the first prototypes of what became the MaKey MaKey during a residency at the San Francisco Exploratorium. These prototypes were hand-soldered boards built around the Teensy board, a derivative of the Arduino open source electronics prototyping platform. The Teensy at that time provided the easiest way to create our own USB HID device to emulate a keyboard and mouse. Using this very first iteration of the design, I ran a small workshop with the Exploratorium Tinkering Studio staff. I was encouraged by the fact that they were able to make a diverse set of functional projects using the prototypes in just a few hours. The projects

in that first workshop included a set of floor pads for playing the game "Dance Dance Revolution" by stepping on them with your feet, an interactive art project that let you make paint splatters on a screen by blowing air across flowers made of foil, a teacup that sensed when you drank from it and sent a tweet, and foil pads attached to a pair of pants that triggered drum sounds. These projects confirmed my intuition that the kit would open up a large space of musical possibilities.
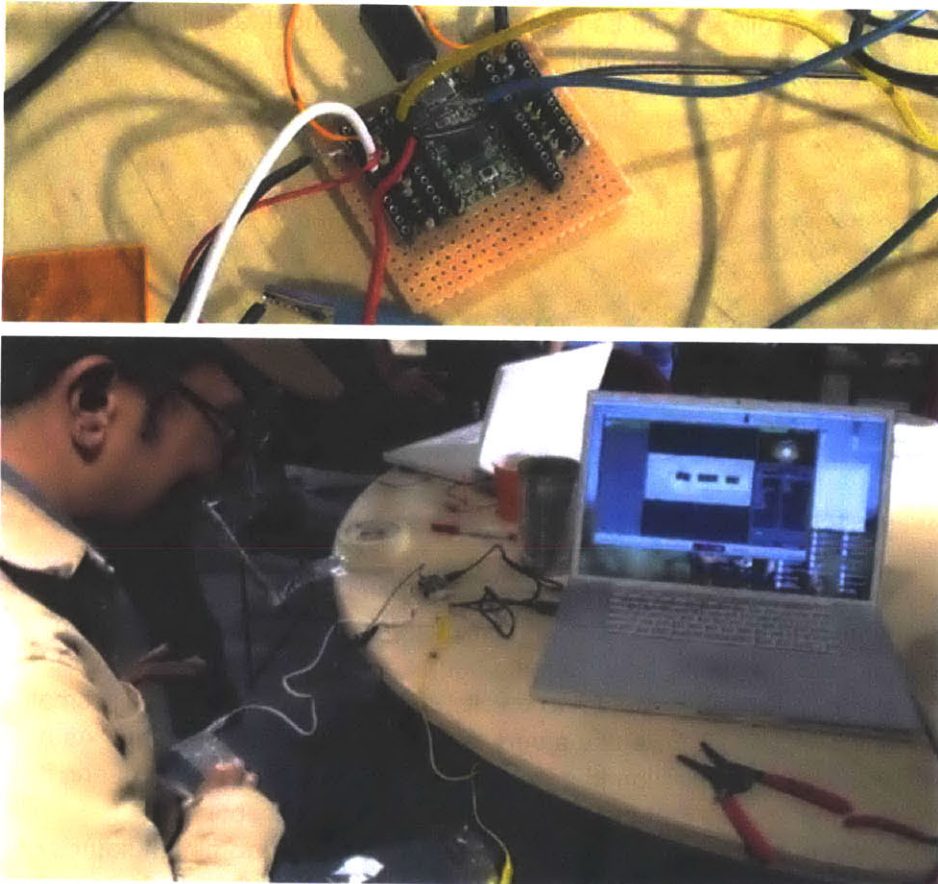
Figure 9: The first prototype of MaKey MaKey. Top: The "Project Q" prototype, made using a Teensy microcontroller. Bottom: Drum pants controlling Garageband music software.

Hardware design

The visual design of the front of the MaKey MaKey circuit board is inspired by the classic game controller from the Nintendo Entertainment System. It has six different locations to connect an alligator clip by inserting it into a pair of holes. The four arrows represent the up, down, left, and right arrow keys. The two circles are for triggering the space key and the mouse left-click. On the bottom is a row of sockets for connecting to ground. On the back of the MaKey MaKey, header sockets provide additional connections. On the left side are

connectors for the w, a, s, d, f, and g keys. On the right side are connectors for triggering mouse left and right clicks, and for moving the mouse cursor on the screen.

Some of these features were chosen to communicate the idea that the MaKey MaKey can be used to create game controllers, and to support that kind of functionality. This is the reason for the visual reference to a Nintendo controller, and the placement of arrow keys, space, and click prominently on the front. It also motivated the inclusion of the "w" key along with the other letters, so that game controllers could be made using w, a, s, and d, a typical alternative to the arrow keys for four-way directional control.
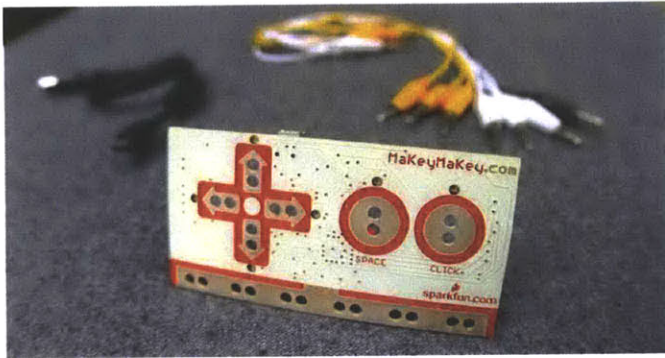
Figure 10: the MaKey MaKey circuit board.

The MaKey MaKey detects input using a very sensitive mechanism which we call "high resistance switching." This system makes it capable of sensing connections through even highly resistive materials such as plants, play-doh and people. Each input uses a typical button circuit, with a resistor pulling the input up to 5 volts, but rather than the typically small value for this resistor, such as 1K Ohm, we use a very large value such as 22M Ohm. As a result, even a weakly conductive connection to ground can pull the input down, triggering it. The problem with this arrangement is that the very sensitive input is susceptible to electromagnetic noise, which can cause it to fluctuate erratically. The predominant source of this noise is the 60Hz "hum" that comes from the AC current supply that normally surrounds us indoors. We minimize the effect of this noise using a simple digital filter that averages the state of the digital input over a moving window with a length the same as one period of 60Hz.

MaKey MaKey and music
In addition to the features that support making game controllers, MaKey MaKey has some features that support making musical inventions. Letter keys from the home row are included because many interactive music software tools and websites use the home row to represent the white keys on a piano. The a, s, d, f, and g keys usually make it possible to play the first five notes in a C major scale: C, D, E, F, and G. These tools are designed to allow the user to play melodies by pressing keyboard keys, but this same functionality enables MaKey MaKey users to construct instruments that play those notes. The MaKey MaKey's quick response is

also useful for music inventions. It has low latency, meaning only a short time interval elapses between the MaKey MaKey sensing an input and triggering a key (about 10 to 20 milliseconds). When it is used to trigger sounds on a computer, this low latency creates a strong feeling of responsiveness.

The MaKey MaKey can be used to make music using a wide range of existing software tools. Several music production environments have a mode that allows the user to play musical notes using the keyboard, including Ableton Live and Apple Garageband. Ableton Live also allows the user to create their own key mapping, so that a keypress can trigger any of the numerous sound controls available on the interface. This means that in addition to triggering individual notes, a MaKey MaKey can be used to do things like trigger sound clips, mute and unmute tracks, and enable and disable audio effects.

Another convenient and flexible way to trigger sounds using MaKey MaKey is with a software tool called SoundPlant. This software shows an image of a computer keyboard layout. The user drags and drops sound files onto the keys, and then pressing a key will trigger its sound. Soundplant also makes it possible to shift the pitch of the sound.

In addition to these music production tools, numerous websites provide music functionality triggered by keypresses and so can be used with MaKey MaKey. These include virtual piano keyboards and synthesizers, drum machines, and sound effects panels.

Figure 11: Music tools that work with MaKey MaKey. Top left: Garageband's musical typing mode. Top right: Ableton Live's key-mapping mode. Bottom left: Soundplant key-triggered sampler. Bottom right: The Virtual Keyboard web app.
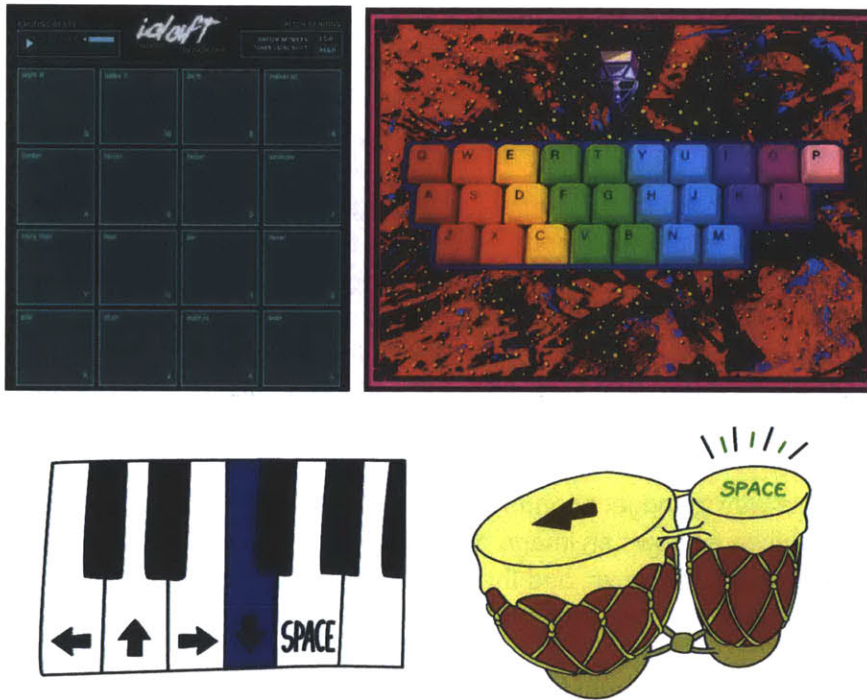
Figure 12: More web-based music tools that work with MaKey MaKey. Top left: a Daft Punk console. Top right: a sound effects panel. Bottom left: a simple piano for use with the front connections on the MaKey MaKey. Bottom right: MaKey MaKey bongos.

The Scratch programming environment (Resnick et al, 2008) is also a popular way to make music using MaKey MaKey. Scratch includes an audio recorder and editor, a library of sounds and music loops, and the ability to play notes and drum sounds using a built-in synthesizer with several instrument sounds. The programming language makes it very simple to trigger an event such as playing a sound when you press a key. This means that people can very quickly use it to create an interactive program that works with MaKey MaKey.

Figure 13: A complete Scratch program for MaKey MaKey. It triggers a meow sound when you press the space key.

We have also created some of our own custom music applications for use with MaKey MaKey, in order to make it easier for beginners to get started. These include web pages for playing bongo drums and piano. Others have created custom music software for use with MaKey MaKey as well, including a recorded piano performance that can be accompanied by triggering a set of notes designed to harmonize with it.

Design Concepts

Several core concepts guided our design of MaKey MaKey. As with MelodyMorph, it is strongly influenced by Resnick's concept of designing for low floor, high ceiling, and wide walls. MaKey MaKey supports low floors by making it easy to get started, high ceilings by making it possible to build complex creations, and wide walls by supporting a variety of themes and interests.

To cover more specific aspects of the design, I will organize a set of design concepts around tinkering loops and tinkering maps, much as I did with MelodyMorph. For tinkering loops, these design concepts are immediate feedback, scaffolded complexity, and authentic context. For tinkering maps, they are flexible space, expressive gesture and linked symbols.

Immediate Feedback

The inner-loops of tinkering involve rapid iterative experimentation, which requires immediate feedback. As Resnick and I describe in our chapter on "Designing for Tinkerability" (2013), MaKey MaKey supports immediate feedback through its low-latency responsiveness. Making a connection on the circuit board immediately lights up an LED, and triggers a keypress on the computer within tens of milliseconds. Depending on the software on the computer, this can trigger both visual and musical output right away.

Scaffolded Complexity

The design of MaKey MaKey and its surrounding materials helps people develop the complexity of their musical projects. As they begin, we support users' first musical experiments with web apps that are designed to make the setup as easy as possible. The piano and bongos pages we created for this purpose have small fixed sets of sounds. These are triggered using only the connections on the front of the MaKey MaKey board, easily accessible with alligator clips. Later, as people progress into more complex musical experiments, they can use the links on our "how to" page to various web sites and software tools. These tools support more connections, including the additional inputs on the back of the MaKey MaKey board, enabling their instruments to play larger sets of notes. Some of the tools are configurable, so users can change the pitches or sounds themselves. If they dive into Scratch programming, they can begin with some examples that we provide, and experiment with more complex behaviors. These include triggering both sounds and visuals, triggering chords and melodies instead of individual notes, and triggering randomized sounds or sequences of sounds. They can also explore more complex physical construction techniques, shifting from using individual items like play-doh and coins to building integrated instruments with organized arrays of connections. The MaKey MaKey comes pre-set with 11

111

different keyboard inputs, but it is possible to use up to 18 keyboard inputs. This requires reprogramming the board using the Arduino programming environment, with simple modifications our open source firmware. As people progress into the complex context-loops of musical tinkering, they can make use feedback from their peers, as they collaborate, share, and perform with their creations. They can seek inspiration from the extensive gallery of projects on the MaKey MaKey website and elsewhere on the web, and ask and answer questions on the MaKey MaKey forums.

Authentic Context

The process of tinkering with MaKey MaKey enters larger tinkering loops as projects evolve over days or weeks. Sometimes these loops include double-loop learning when the project raises questions about the tinkerer's identity and larger role in the world. In order to support these longer-term projects and larger tinkering loops, the tool must enable meaningful connections to the tinkerer's own musical and social worlds. With MaKey MaKey these connections can include incorporating physical materials and creating physical designs that you care about, and customizing or creating your own sounds. It can mean making music that reflects your own musical culture, and even playing your invented instrument along with the traditional instruments of that culture. It can even involve performing for an audience, or creating a music video, using a MaKey MaKey instrument.

Flexible Space

To support the creation of tinkering maps, which display externalized traces of musical thinking, the tool must support the flexible use of space. This flexibility enables you to express your own unique way of thinking about musical structures. MaKey MaKey musical instruments are made up of discrete objects that each trigger a note. These objects can be positioned anywhere in space relative to each other. People can build a row of objects that make a musical scale, or a row in the order of a melody. They can position their objects to mimic some acoustic instrument, or to make the instrument easier to play. Or they can come up with some other way to think about their spatial arrangement. MaKey MaKey works with a wide range of materials, so the objects themselves can be large or small, soft or hard, and vary in color, texture, or other properties. The inputs can be made from food or even living creatures. MaKey MaKey instruments can also be made using graphite or conductive paint, which can be used to create complex spatial arrangements and musical maps.

Expressive Gesture

MaKey MaKey supports the creation of meaningful tinkering maps by supporting intuitive and expressive creative processes. The fundamental way in which MaKey MaKey instruments are expressive is that their immediate responsiveness allows you to play a sound by triggering it with a tapping gesture or body movement. One limitation of MaKey MaKey is that its discrete sensing does not allow continuous control, so it cannot detect how hard you tap something and use that control the loudness of the sound. But variation in musical timing can be as expressive as variation in loudness.

Linked Symbols

To make tinkering maps with the property of "linking," they must connect symbol and action. This is a fundamental aspect of the functionality of MaKey MaKey, linking together a physical object with a sound played by the computer. The choice of both object and sound is very flexible, and so it is possible to create a wide variety of arbitrary relationships between object and sound.

Design challenges

The challenges we faced in designing MaKey MaKey included making it easier for users to make physical connections, and making it easier for them to map keys to sounds.

Making physical connections

The two types of connections provided by the MaKey MaKey circuit board each presents its own challenges. The front of the board has pairs of holes designed as sockets for the alligator clips that come with the kit. The clips are meant to be connected by squeezing them to open their jaws and putting one side into each hole. The jaws can be difficult to open, because of the amount of force required, and the slipperiness of the insulating sheath. In general, these sockets seem easy enough to use, but the alligator clip jaws make them difficult for young children and those with limited hand strength or dexterity. One solution to this problem would be a non-standard clip which is larger and requires less force to open.

The back of the MaKey MaKey has header sockets designed for plugging in solid-core wires. These sockets are awkward to use for several reasons. The wires pull out of their sockets more easily than alligator clips. People typically attach an alligator clip to the end of each wire, and then use the alligator clip to connect to other materials. This additional connection makes their construction more complex and the connections more difficult to trace and debug. Ideally, we could have a single robust connector type for all the connections. The header sockets are a tradeoff that allows us to add more inputs in a smaller area of the circuit board.

Placing these additional connections on the back of the MaKey MaKey hides them from beginning users, which helps keep the initial experience simple. But the tradeoff is that some people do not notice that there are additional connections. It is also awkward to combine the two types of connections. Sometimes people plug alligator clips into their sockets from the back of the board. This works, but can be confusing because of the lack of labels for those connections on the back of the board, and the fact that the left and right arrows are reversed.

People invent their own creative ways to make these physical connections to the circuit board when they run into problems. People who have trouble opening the alligator clips sometimes attempt to jam the closed alligator clip into a single hole, which can just barely made to work, though it easily falls out again. I have also observed people attempting to connect alligator clips to the header sockets, clipping onto the side of the circuit board to make a connection with the tiny feet of the header. They also put the clip over the top of the header, so that the

tips of the clip touch the feet of the header. These connections also just barely work, though they are easily dislodged. The fact that people are forced to invent these partial solutions suggests that we need connectors that are easier to use.

Mapping keys to sounds

The usefulness of MaKey MaKey for making music is limited by the user's ability to map keys to different sets of sounds. Both the arrangement of inputs on the board and the limitations of existing software can make it difficult for users to change this mapping.

The front of the MaKey MaKey, designed to emphasize its use for video games, provides connections for the arrow keys and the space key. These keys are not typically used to trigger sounds in music software. We provide custom interfaces to play piano notes and bongo drum sounds using these keys, but it is not obvious how users can find an easy next step. One path is to use the connections on the back of the MaKey MaKey, but even these provide only six more keys. And the five home row keys available on the back can be used to control many existing piano interfaces, but can only play a fixed set of five pitches. It would be much easier to play more complex melodies using MaKey MaKey if it came with a set of sockets on the front that mapped to the 13 keys corresponding to an entire octave of the home row piano.

The musical flexibility of the MaKey MaKey also relies on the accessibility of reconfigurable music software. Many people do not have such software, or if they do they may not have sufficient knowledge to reconfigure it. Generally, it is not straightforward to change the pitch triggered by a particular key in most music software. Some exceptions include Scratch (which requires writing a simple program that causes a key to a note), Ableton Live (expensive professional software), and Soundplant (free, but the user must provide their own sound samples, and the pitch adjustment slider moves in quarter tones, an atypically small pitch interval). A useful future project might be an interface specifically designed for MaKey MaKey that makes it easy to change the pitch of a note triggered by each key.

Related Work

There are a few other projects that aim to provide a kit for making physical-digital musical instruments. Here I review the MICK musical instrument construction kit, the Performamatics project, Satellite CCRMA, Ototo, and TouchBoard.

MICK musical instrument construction kit

The MICK musical instrument construction kit (Thibault, 2001) is a research project that combined software on a Pocket PC mobile device with a custom hardware device connected to a set of sensors for creating music controls. The hardware was initially developed using the Cricket platform described above, and later transitioned to the Tower platform (Gorton and Mikhak, 2004). The software on the Pocket PC included functionality for mapping sensor input to output, such as playing a synthesized musical note, a recorded sound, or a melody. The interface included different ways to map sensor input to musical notes. The user could define a range of sensor values inside which the instrument would respond. Within this range, they

could choose from different functional mappings that mathematically transformed the sensor value into musical output in different ways: linear, inverse, square, and square root.

Two MICK project examples are shown in the figure. One is a simple guitar-like instrument with a body built from Lego bricks, and several mechanical switches that are used to trigger notes. The two switches on the neck of the guitar did not play notes themselves, but instead modified the pitches triggered by the three switches on the body, simulating the effect of frets on a guitar. The other example is a sliding car instrument. A small car on raised Lego rails can slide back and forth over a row of stacked lego bricks. A sensor in the car measured the distance down to the stack of bricks, and then a program transformed this sensor input into pitches, so that taller stacks produced higher pitched notes.

Figure 14: MICK musical instrument construction kit. Left: a guitar with switches. Right: a sliding car with a light sensor.

MICK has a more flexible mechanism than MaKey MaKey for sensing, including continuous sensors of several types. It also has a flexible way to map input to output, using programmable functions. MaKey MaKey's design trades away this kind of flexibility for simplicity and accessibility.

Performamatics

The Performamatics project (Greher and Heines, 2014) is an interdisciplinary approach to teaching computer science at the undergraduate level combining programming, music, and physical computing. The Performamatics curriculum makes use of the Scratch programming environment for making music. With the addition of the Scratch Sensor Board or a variant called the IchiBoard, students can control Scratch using physical controllers of their own design. Two examples of Performamatics projects are shown in the figure. One student built an electronic tympani drum. He attached the IchiBoard to the stand for a drum head, and used its microphone to detect a spike in the sound level when he hits the drum. At the same time, he is pulling on a string connected to a slider on the IchiBoard. The Scratch program plays a sound each time he hits the drum head, with a pitch controlled by the slider. Another student created a performance using the resistance sensors on the IchiBoard. She connected

alligator clips to an array of fruits and vegetables, and also to another person's face, and used the resistance measurement to control the pitches of a stream of notes.

Figure 15: IchiBoard instruments from the Performamatics project. Left: a drum-head controlling pitched percussion sounds. Right: musical fruit.

Like MICK, the Performamatics project focuses on programmable relationships between sensing and sound, especially creating mappings between continuous sensor values and pitch or loudness. These mappings are less encapsulated than in MICK, because they are computed using Scratch programs. But in practice the actual mappings used are simpler, because of the difficulty of expressing non-linear relationships between input and output, and of mapping to musical scales, using Scratch.

Satellite CCRMA

The Satellite CCRMA project (Berdahl and Ju, 2011) is a platform for constructing musical instruments that is designed for use in a university-level music interaction design context. It is based around an embedded computer running a customized version of the Linux operating system. It includes several programmable music tools, primarily for sound synthesis using tools like Pure Data and ChucK, and an Arduino-compatible microcontroller, primarily for collecting and processing sensor data. Examples of student projects made with this platform include a slide-guitar-like ribbon controller box, a camera-based system that synthesizes sound by analyzing patterns of waves in the surface of a tub of water, and a set of stompboxes that mimic and extend the typical functionality of guitar effects pedals to transform sound.

The creators of Satellite CCRMA focus on priorities including longevity, autonomy, and compactness. With these properties, a musical instrument made with their system can last many years, operate on its own without an external computer, and be small enough to store easily. They also prioritize extensibility of the system, so that others can add to the platform. Each of these properties makes a trade-off with accessibility. MaKey MaKey is on the opposite end in each case. MaKey MaKey instruments are likely to be short-lived, only work with a computer, and are not necessarily easy to store. It is not easy to extend or modify the

behavior of the MaKey MaKey board, relatively speaking. It is based on the Arduino open-source electronics platform, which is designed around accessibility, but requires a level of technical skill above that of the typical MaKey MaKey user. As a result of these trade-offs, the MaKey MaKey is quick and easy to set up, and works conveniently with the existing computers that people already have. Its focus on creative tinkering means it supports quick experiments, instead of more permanent construction.

Ototo and TouchBoard

More recently, two new products have emerged that also support creation of physical-digital musical instruments. These are the Ototo from Dentaku, and the Bare Conductive TouchBoard. They emerged as commercial products after MaKey MaKey, and appear to be inspired by it. They are both too new to give detailed descriptions of projects created by users, or of the pedagogy around them.

Both products use capacitance sensing, as opposed to the high-resistance switching technique we designed for MaKey MaKey. Capacitance sensing can detect when a human body comes in contact with the circuit, and works with objects of fairly high resistance. The person triggering it does not need to be connected to ground, as in the case of MaKey MaKey. In practice, this means the set of physical materials one can use is quite similar to the set for MaKey MaKey. It does constrain the types of connections that can be made. In table 1 of the MaKey MaKey case studies chapter, I list several connection types that are possible. With capacitance sensing, the skin-to-object connection is possible, while skin-to-skin is probably not possible. The tool-to-object, bridge and switch connections may or may not be possible, depending on how the capacitance sensing is set up.

Both Ototo and TouchBoard have 12 inputs, which are mapped to musical notes. They both advertise some sort of MIDI output capability, as well as on-board sound synthesis. The TouchBoard has an SD card socket, for loading sounds files that can be triggered by the inputs.

The Ototo is more explicitly designed for music, with a piano keyboard design indicating the pitch for each connection. The use of 12 chromatic notes creates an incomplete scale, because it does not reach the top note (i.e. it starts on C and ends on B). The Ototo also has connectors for a set of several different continuous sensors, including a slider, knob, light sensor, force sensor, and wind controller. These enable it to combine discrete note triggering, as in MaKey MaKey, with continuous control of musical parameters such as loudness or filter frequency.
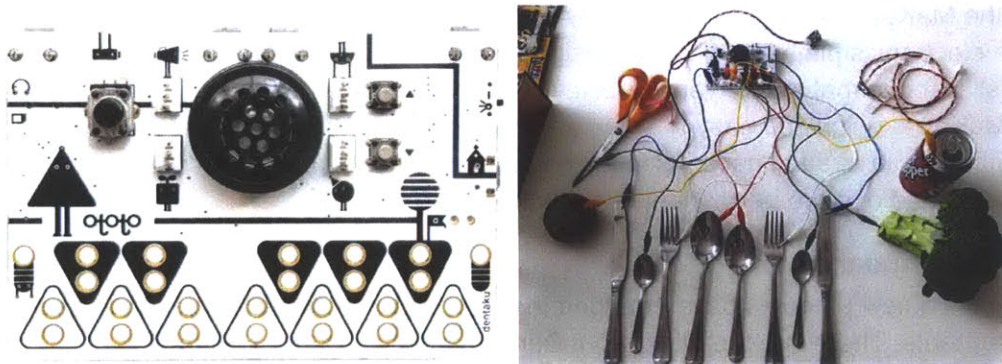
Figure 16: The Ototo sensor board. Left: the circuit board with 12 capacitive sensing inputs mapped to chromatic notes. Right: Ototo connected to various conductive objects (images by Dentaku Ltd).
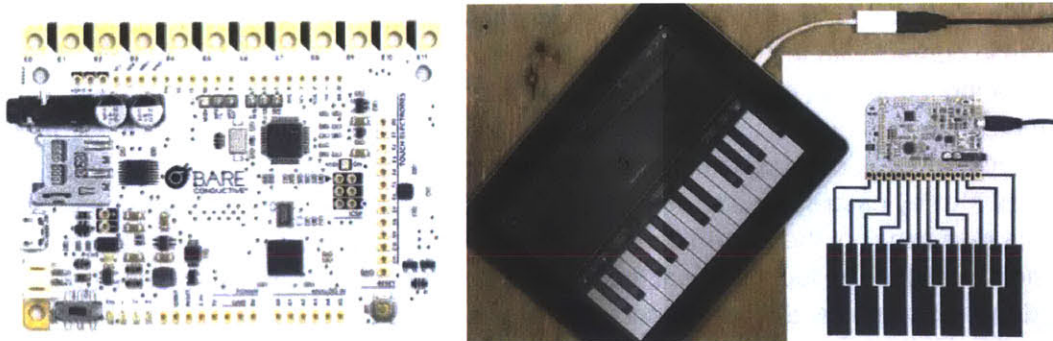
Figure 17: The TouchBoard sensor board. Left: the circuit board, with 12 capacitance sensing inputs along the top. Right: TouchBoard connected to a painted piano to control an iPad music app (images by Bare Conductive).

Future work

We designed the MaKey MaKey invention kit to support as wide a variety of projects as we could, including music but also video games and other inventions. It is possible to imagine a future version of MaKey MaKey with functionality more specifically focused on musical tinkering. The recent projects Ototo and Touchboard are similar enough to this idea that they act as experiments in this space of possibilities.

The use of capacitance sensing is appealing, because it removes the necessity of grounding the player's body. One trade-off is that this technique does not support the popular MaKey MaKey human-synthesizer projects that are possible with high-resistance switching, and it may not support other non-skin connection types. Another tradeoff is that the standard capacitance sensing chip has twelve connections. It seems desirable, for flexible musical instrument construction, to have at least 13, so as to have a full chromatic scale, and ideally many more.

Ototo's variety of continuous sensor inputs is also appealing, but these support less flexibility in construction than the discrete sensor inputs, which can be attached to so many different materials. It may be possible to use a technique like Drawdio's high resistance sensing to create continuous inputs that can be controlled using a wide range of materials.

The use of MIDI, and on-board samplers and synthesizers, give the Ototo and TouchBoard potentially much greater musical flexibility and autonomy than the MaKey MaKey has. One question about MIDI output is whether this is easy enough for novices to get started with. The keypress output from MaKey MaKey is much more familiar and friendly to novices. The advantages of MIDI may make it worth the shift, or it may be possible to create a hybrid device that has both MIDI and keypress output.

A wireless MaKey MaKey would open up a few interesting possibilities, especially for carrying instruments around without being tethered by a USB cable to the computer. The low energy bluetooth protocol has low enough latency to be used in music controllers, and also makes it easy to connect to a mobile device rather than a laptop. A more radical shift would be to make the individual sensor inputs themselves into separate wireless devices. This would make it possible to distribute inputs much more easily around the body, furniture, room, playground, or even larger spaces.

Another avenue to consider is branching into specialized kits for different types of musical tinkering. One kit might especially support hand-held mobile musical instruments. Another might focus on attachment to paper for musical paintings, or to fabric for wearable instruments. Finally, it is worth considering output in the form of actuation, rather than sound. This would mean controlling an array of motors, such as servos or solenoids, that can be attached to physical objects in order to tap or shake them to make sounds.

Chapter 5:
MaKey MaKey Case Studies

In this chapter I will review some examples of musical tinkering with MaKey MaKey, including documentation of projects "in the wild," notes on my own workshop experiences, and stories from music educators using MaKey MaKey in classrooms.

Musical Tinkering in the Wild

At the time of this writing, fall of 2014, approximately 100,000 MaKey MaKey kits have been distributed around the world. These kits have come into the hands of people using them in a wide range of contexts. We designed the kit with a focus on open-endedness and accessibility, hoping to reach a broad audience. The kit has been used by children, families, and adult novices in their homes; teachers across K-12, undergraduate and graduate level education; informal educators in museums, libraries, after-school programs, and maker spaces; researchers in fields like interaction design, physical computing, and industrial design; occupational therapists and others designing customized assistive technologies; and professional designers working with museums and advertising agencies. In many of these situations, the kit was used to make music. We see evidence of musical tinkering across these contexts, especially when it is used in music classrooms, by amateur musicians augmenting their existing instruments or designing new ones, and by professional musicians using it in music videos and live performances.

The users of the kit have also generated a large amount of social media activity, including thousands of tweets and youtube videos, and hundreds of vimeo videos, vine videos, and blog posts. MaKey MaKey can be used for activities outside of music, but a large fraction of all of this content (difficult to estimate) is music-related. To help users of MaKey MaKey get a sense of the space of possibilities, we have created a curated gallery on the site that collects documentation of user-created projects from around the web. At the time of writing, this gallery contains 185 projects. Of these, 110 are tagged "music" (the proportion of music projects within the gallery is biased, because I have been curating the projects and I am especially interested in music).
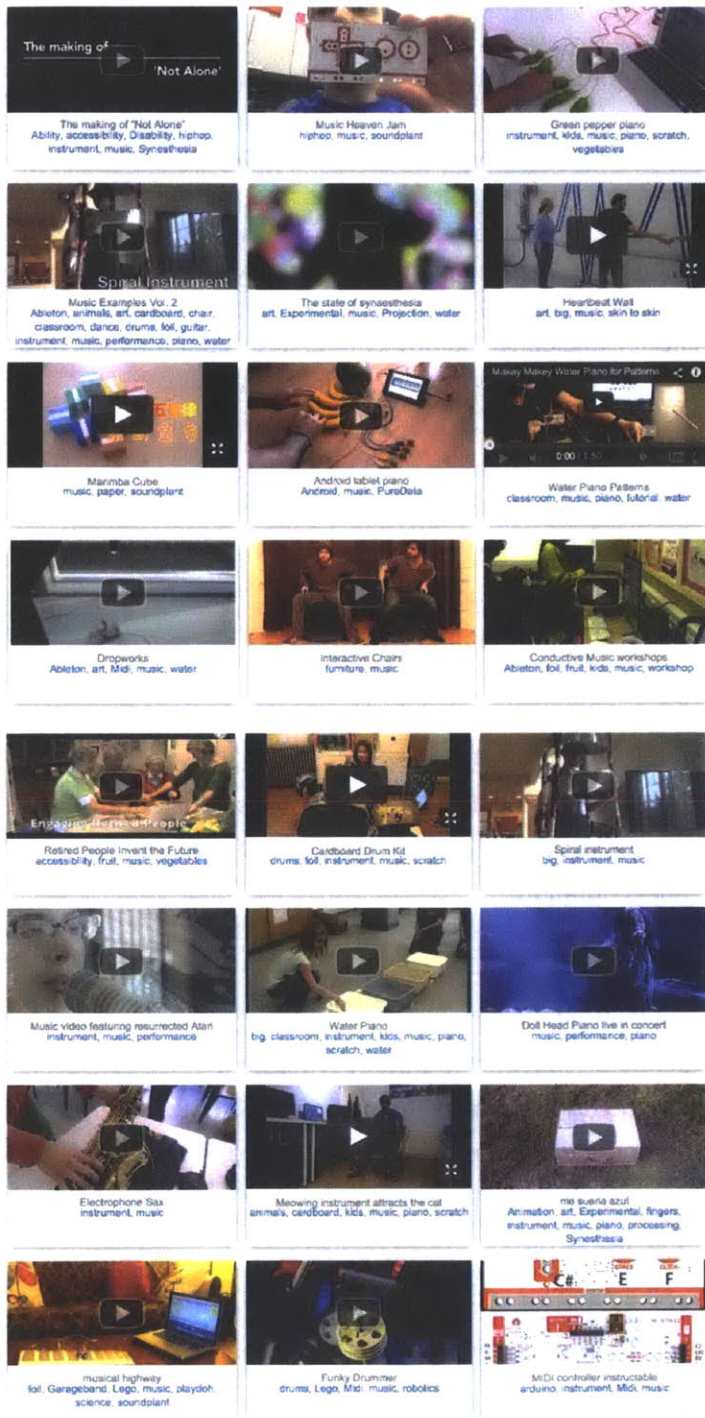
Figure 1: The MaKey MaKey website's project gallery showing 24 of the 110 projects tagged "music."

A catalog of music projects

To help make sense of the diverse set of MaKey MaKey music projects appearing in the wild, I have created a categorization scheme. I describe the projects using three different attributes: instrument family, pitch arrangement, and connection type (see table 1 below). For each attribute, I describe several distinct categories. These three attributes and the categories within them emerged from my experiences looking at large numbers of MaKey MaKey music projects on social media sites and noticing patterns among them. My intention is to make the sets of categories complete, so that any MaKey MaKey music project could be described using them. The categories are not mutually exclusive, so any particular project may fall into more than one.

INSTRUMENT FAMILY

Imitation	Visually and/or musically similar to an existing instrument
Augmentation	Adds functionality to an existing instrument
Automation	Instrument plays itself
Invention	Novel or invented instrument paradigm

PITCH ARRANGEMENT

Piano-like	Row of pitches ordered in a row
Other-instrument-like	Pitches organized like a string, wind, or percussion instrument
Melody-specific	Set of pitches designed for playing a particular melody
Score-like	Playable music notation

CONNECTION TYPE

Skin-to-skin	Human touches a human
Skin-to-object	Human touches an electrically conductive object
Tool-to-object	A tool held in the hand touches a conductive object
Bridge	Use skin or conductive material to bridge between two contacts
Switch	Simple mechanism completes a circuit

Table 1: Categories of MaKey MaKey music projects

Instrument families

MaKey MaKey musical instruments fall into four families: imitation, augmentation, automation, and invention.

Imitation

These projects imitate existing acoustic musical instruments, by making reference to their visual and musical properties.

The "banana piano" is a common introductory MaKey MaKey music project (and is shown in the main MaKey MaKey video). It consists of a row of bananas lined up like piano keys, with each banana triggering a piano note. The pitches ascend from left to right as on a piano. Pianos made from rows of conductive objects are popular. These also include pianos made from various other fruits and vegetables, flowers, gummy worms, sausages, and so on. The floor piano, as shown in the figure, is another common variant, as are piano stairs and piano ladders.

Figure 2: Pianos. Left, a banana piano. Right, a floor piano played with the feet.

MaKey MaKey drum kits take a variety of forms. The "mini-drums" is a tiny drum kit. In one case, this was made of paper, foil and play-doh, playable with finger-tips. In another case (shown in the figure below) it appears to be built by adding foil connections to an existing toy drum kit. The "cardboard drum kit" is a life-size drum kit made of cardboard and foil, played with a set of what appear to be foil-covered spoons.

Figure 3: Imitation drum kits. Left: A miniature drum kit played with tiny sticks. Right: a life-size cardboard drum kit.

Similarly, MaKey MaKey guitars range from the miniature "spatula guitar," made from a novelty spatula in the shape of a guitar, to the "card guitar," which is roughly life-size. These are played by touching a row of contacts along the neck, and have no representation of individual strings. They are shaped like, and can sound like, an actual guitar, but they do not work much like one. Instead, both examples function more like the plastic controller from the video game "Guitar Hero," which has a row of buttons on the neck that can trigger different guitar notes.

Figure 4: Imitation guitars. Left: the spatula guitar. Right: the card guitar.

Other imitation instruments include the cardboard trombone and the plastic tube clarinet. The trombone has a cardboard slide that mimics the role of an acoustic trombone's slide, making the instrument play lower pitches as the slide extends outward. But it plays a diatonic scale rather than a chromatic scale. It also has no system to mimic partials, the way a brass player uses her lips to move the pitch up and down into different ranges. Its cleverness lies in the construction of a mechanism that adapts the trombone's continuous slide to the discrete world of MaKey MaKey, making different connections along its length. The clarinet, below, is actually played by touching it to your face, but this connection does not provide a way to control the instrument. Instead it works as a clever trick for grounding the player, using a metal contact that touches the skin.
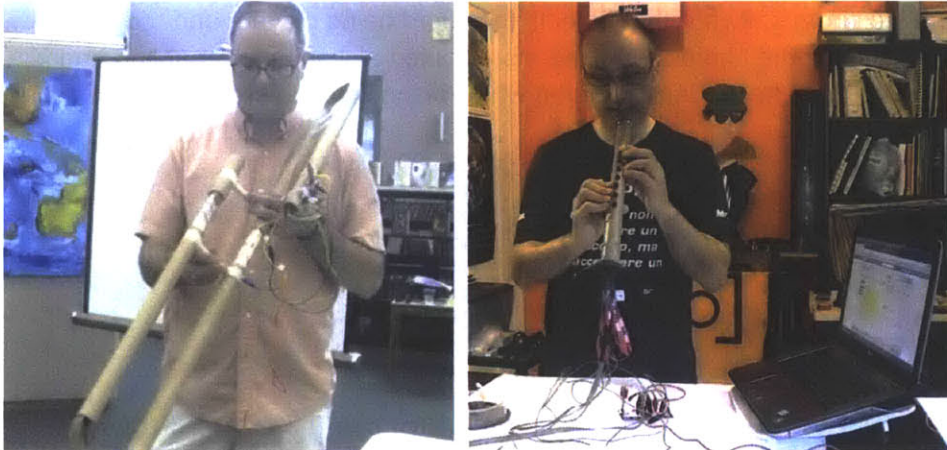
Figure 5: Imitation wind instruments. Left: a cardboard trombone. Right: a plastic tube clarinet

Augmentation

These projects add functionality to an existing musical instrument. Many of these add buttons to the surface of the instrument to trigger additional notes, often using a distinct instrument timbre. The first guitar shown in the figure adds notes using conductive tape. The second guitar has large coins attached that trigger bass guitar notes. The third instrument is a kora (a West-African stringed instrument), with the the decorative thumb tacks in its resonator gourd connected to trigger sounds of a Hang drum (a resonant pitched steel drum).

Figure 6: Augmented instruments that add notes. Left and middle: notes added to the guitar body. Right: a kora with notes added to thumbtacks on the resonator gourd.

Some augmented instruments integrate other functionality into the instrument, such as control over music production software. In the "Lingua Looper" project, a sitar is connected to computer with looping software, which can record, repeat and layer sounds from the instrument. Because the sitar requires both hands to play, and sitting with it constrains the feet, the inventor triggers the looper by touching her tongue to a hanging metal ball. The grand piano in the center of the figure below has metal pads above the keys, and a metal strip below the keys, for controlling looper functionality without moving hands away from the piano. For the "Key Effect" project, on the right in the figure, the student opened up the piano and added contacts to the hammer mechanisms so that the computer could detect which keys

125

were being pressed. He wrote software that detects the chord being played, and triggers a recording of a person singing the chord name. He also created a system for enabling and disabling delay and distortion effects depending on which piano keys he presses.

Figure 7: Augmented instruments with added computer functionality. Left: The "Lingua Looper" augmented sitar. Middle: An acoustic piano with looper controls. Right: The "Key Effect" piano, which sings chord names and adds effects.

Automation

These MaKey MaKey projects include a mechanical system that causes the instrument to trigger itself, in order to play notes in a particular pattern or sequence. Some use vehicles. The musical highway and musical train in the figure below make use of a small vehicle moving along a track. When the vehicle drives over each metal patch on the track, it triggers a note. The musical highway plays the first phrase of "Twinkle, Twinkle Little Star," and the train plays a more arbitrary sequence.
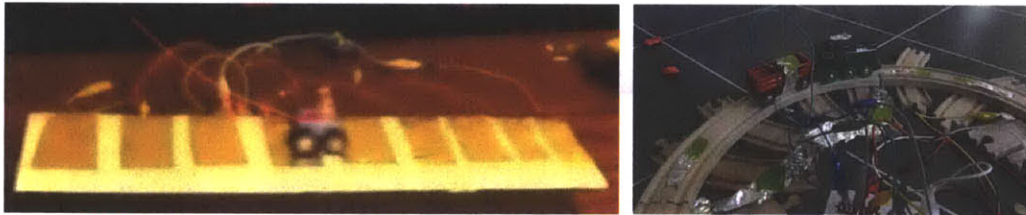
Figure 8: Musical vehicles. Left: a musical highway that plays "Twinkle, Twinkle." Right: a musical train.

Other automated instruments use rotary mechanisms. On the turntable in the figure below, the carrot touches wires sticking up around the edges as it rotates, triggering percussion sounds in a rhythmic pattern. In the complex Lego mechanism in the figure, a paper disc with foil strips laid out on it rotates beneath several arms with safety needles attached. The safety needles drag along the paper, and complete a circuit when they touch a foil strip, triggering a percussion sound. The arms can be manually moved in and out, in order to reach foil strips of different lengths, which control the rhythmic pattern.

126
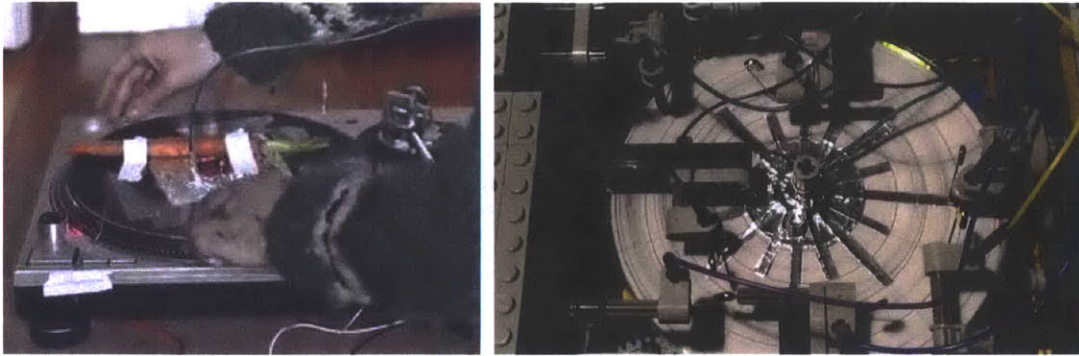
Figure 9: Automated instruments with rotary mechanisms. Left: a turntable with a carrot that rotates, touching wires to trigger sounds. Right: a Lego radial sequencer, with needles that touch foil arms as the disc rotates, triggering rhythmic patterns.

Invention

This category includes MaKey MaKey musical instruments that do not fit any existing paradigm, instead representing clever inventions. Some of these make use of existing tangible interactions with objects. In the figure below, the "Marimba Cubes" is a toy inspired by an origami design that triggers marimba notes when you touch the foil tape. The cubes can fold together, causing the foil segments to touch either. In that case when you touch one of them, multiple notes trigger together, playing a chord. The "Beat Balls" project, below, makes music from juggling. The rubber gloves have foil on two of the fingertips, and the ball is covered in foil as well. When the juggler catches the ball, it bridges the connection between the two fingers, triggering a sound. The custom software for that project triggers different percussion sounds each time a ball is caught or thrown by either hand.
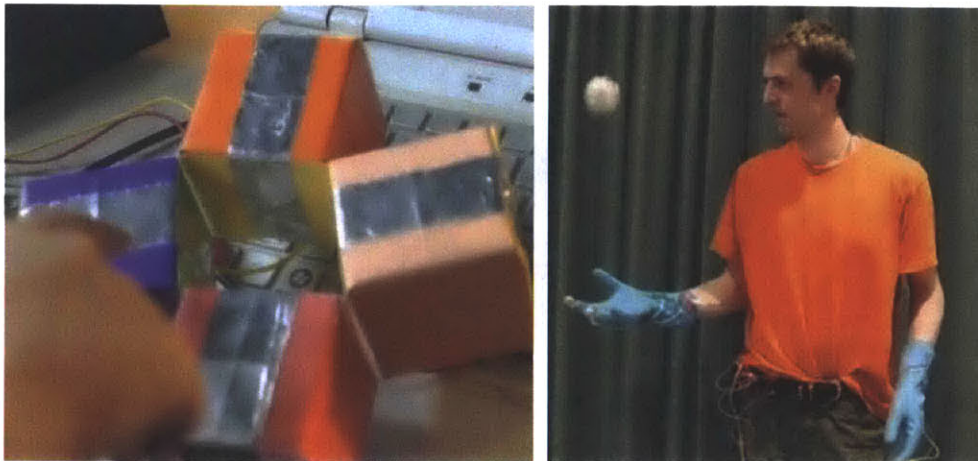
Figure 10: Invented object-scale instruments. Left: Marimba Cubes that play individual notes when touched, or chords when folded together. Right: Beat Balls is a juggling system that triggers a note each time a ball is caught or thrown.

Some musical inventions make use of furniture and other large structures. The "Musical Chairs" project uses chairs mounted on discs that can tilt as the person rotates their chair. There are contacts attached to the discs that touch each other depending on which direction the chair is tilted, triggering sounds. People spinning and tilting their chairs in synchrony can make coordinated musical patterns. In the "Spiral Instrument," the player stands inside of a sheet of foil thats makes a spiraling tube around his body. By touching different contacts attached to the foil, he mutes and unmutes different tracks in an electronic music composition.

Figure 11: Invented furniture-scale instruments. Left: Musical chairs that trigger different sounds depending which direction they tilt. Right: A spiral instrument that surrounds the body with contacts to mute and unmute music tracks.

Pitch arrangements

Another way to categorize MaKey MaKey musical creations is by the way in which they arrange musical pitches in space. Many of the instruments are either organized like a piano, with pitches in a scale lined up in a row, or patterned after some other existing instrument. Some instruments incorporate a set of pitches specific to a particular melody, making them specialized to play that one tune. Other projects have pitches arranged like musical notation, so that they become playable musical scores.

Piano-like

The "imitation instrument" family above includes numerous piano-like designs. Most commonly, these pianos are simple, playing only the first six notes of a major scale. Beginner projects are often limited to six or fewer notes because these correspond to the six alligator clip connections available on the front of the MaKey MaKey circuit board. The assignment of these six connections to pitches is limited by the available software options with predefined mappings. These include a web page on the MaKey MaKey site, and a simple Scratch project, both of which map the six keys to the first six notes of a C major scale.
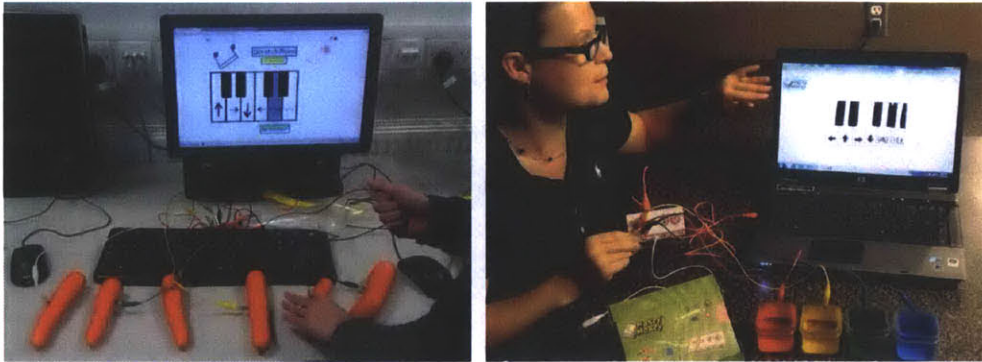
Figure 12: Simple pianos. Left: Carrots connected to six notes in a major scale, using a Scratch project. Right: Cups of colored water connected to four notes in a major scale, using the piano page on the MaKey MaKey web site.

More complex pianos created with MaKey MaKey usually have up to11 notes. This is because, of the 18 total connections available on the MaKey MaKey, 11 of them trigger keyboard keys (the rest trigger mouse clicks and mouse movement). There are various strategies for mapping the full set of 11 keys to different pitches, including making a custom Scratch project, using the Soundplant software for mapping keys to sound clips, using MIDI-mapping software tool that converts key presses into MIDI notes, using music software with a configurable mapping between keys and notes, and creating custom software.

Figure 13: Complex pianos. Left: A piano made using 11 bananas, mapped to vibraphone notes in a minor scale. Right: A piano made using 11 vegetable chunks, mapped to an idiosyncratic scale (C major, plus C#, F# and A#).

Other-instrument-like

Beyond the piano metaphor, some MaKey MaKey instruments use spatial arrangements of pitch or sound that are inspired by other acoustic instruments. In many cases, projects that imitate other acoustic instruments simplify their pitch arrangements, as in the guitar, trombone, and clarinet examples above. Rather than imitating the more complex arrangements of strings and frets on a guitar, or the slide and partials on a trombone, these

often simply use rows of diatonic notes. But the drum kit projects, for example, do generally imitate the spatial arrangement of sounds in acoustic drums.

The kalimba project in the figure below is one example of a careful reconstruction of the spatial arrangement of an acoustic instrument. The kalimba (also known as a thumb piano or mbira) is an African instrument with tuned metal tines which are played by plucking with the fingers. There are many variations of the arrangement and tuning of the tines. They are commonly arranged in two rows that converge in a vee shape, with the lowest pitches in the center. The instrument is played with two hands, each playing the tines on one side. A typically pitch arrangement forms a scale ascending outward from the center with successive notes alternating sides. The scale can be played quickly by plucking on alternate sides. The MaKey MaKey version shown recreates this spatial arrangement, on a box with two arcs of metal tabs that can be triggered by the left and right thumbs. The Scratch project on the computer screen behind it shows an alternating pitch arrangement for a pentatonic scale. The Scratch project also cleverly makes the kalimba reconfigurable, so it is possible to switch between different scales, keys, and timbres, extending it beyond the capabilities of an acoustic kalimba.
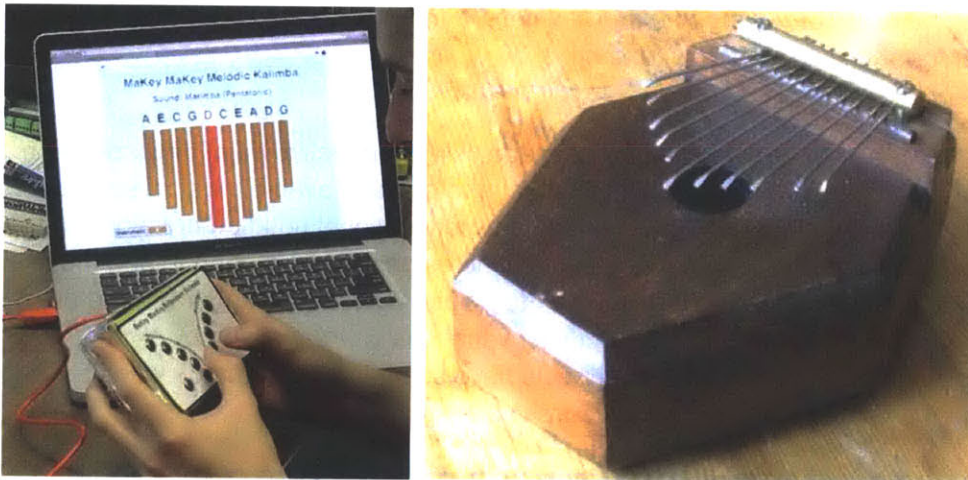
Figure 14: A reconfigurable kalimba. Left: The kalimba's notes are triggered by tapping metal tabs with the thumbs. A Scratch project shows its current pitch arrangement. Right: an acoustic kalimba, played by plucking tines in a similar vee shaped arrangement.

Melody-specific

Some MaKey MaKey instruments are designed around a specific melody. The repurposed Atari instrument shown below was developed by a music group in order to perform a particular melody on one of their songs. They show the instrument being played in the music video for the song. It is made from the case of an Atari video game console, replacing its five buttons with foil contacts for triggering five different notes using a toy piano-like sound. The notes are B, D, E, F# and G. This set of pitches does not match a typical scale, but is instead

chosen to allow the instrument to play the melodic line in one particular song. The doll head piano, similarly, is designed for performance of a melody in a single song, in this case in live performance. The instrument consists of a row of seven doll's heads, each with a pair of contacts positioned on top which can bridged by a hand to trigger a note. In this case the notes are A C E F G C' (one head is unused). The lead vocalist of the band plays a repeating melody on the doll heads while she sings.
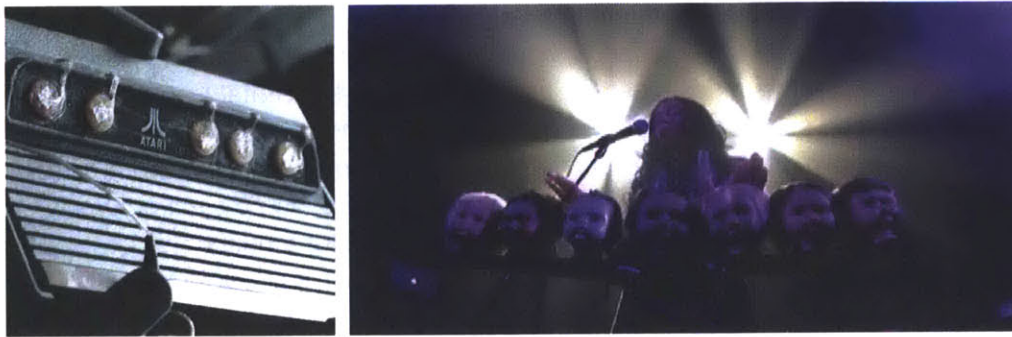
Figure 15: Melody-specific instruments. Left: An Atari video game console, repurposed to play the melody of a particular song in a music video. Right: A piano made of doll heads, made for live performance of a particular song.

Score-like

The connections to MaKey MaKey inputs can be made using conductive paint or graphite on paper. Some MaKey MaKey projects use this technique to create playable musical scores. Some are free-form drawings, that do not resemble any conventional notation or layout. In the figure below, one of these is a ring with a sequence of notes positioned around it, and the other is a swirling pattern with each arm connected to a note. By sliding a finger around on either drawing, you can play a melody. You can play variations on the melody by sliding quickly or slowly, forward or reverse. These projects have something in common with the musical landscapes I described in the MelodyMorph case studies chapter.
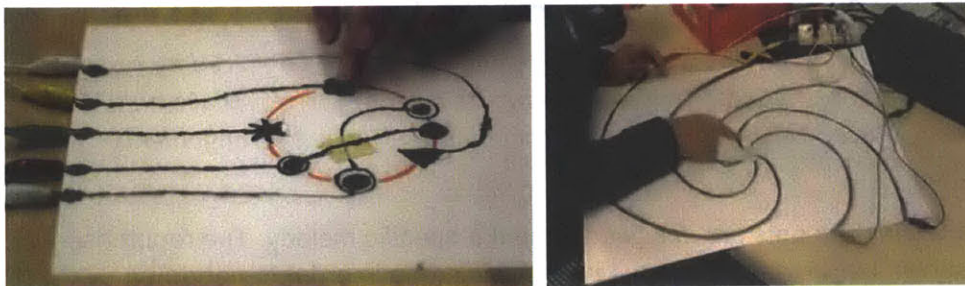
Figure 16: Playable score drawings. Left: a conductive paint ring melody. Right: a pencil swirl.

Other projects use a grid pattern to create a playable score. The first example in the figure contains the melody to the song "When the Saints Go Marching In." It is designed to be played by touching the boxes one at a time, left to right across each row, top to bottom. This notational system represents only the order of the notes, and not the rhythms of the melody. It is constructed using just five different connections for the five pitches that make up the entire melody. The rows start out using them in order, but as the melody changes, the creator of the project had to deal with the increasingly difficult problem of connecting them to each other without crossing lines (the ring melody in the figure above shows one work-around for this problem, using a piece of masking tape to allow one connection to "jump" over another without touching it). This problem is essentially the same as the problem of routing traces on printed circuit boards, which for complex boards is done with the help of algorithmic automation.

The other project in the figure below is designed to play the first few phrases of Strauss' "Also Sprach Zarathustra," also known as the 2001 theme. Because it is also laid out on a grid, one might imagine that it is also played left to right and top to bottom. But in this case, it is designed to play both melodic lines and chords. To play the theme, you start by playing the notes in the filled boxes in the first row one at a time, left to right. Then you play a chord by pressing all notes in the second row simultaneously. Then you play the next chord on the third row, and continue with the melodic line left to right in the next row, and so on.
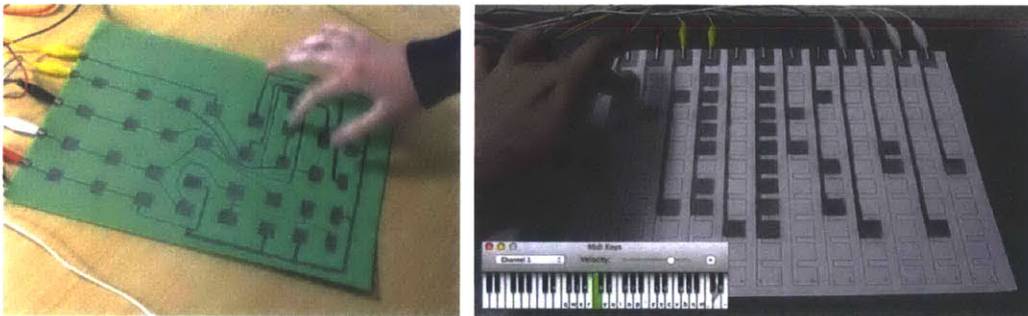
Figure 17: Playable score grids. Left: "When the Saints Go Marching In." Right: The 2001 theme.

These last two playable scores are played with a tool. In the first one, a foil wrapped implement is touched to dots of conductive paint one at a time. Arrows drawn in pencil show the sequence necessary to play the melody from the song "Lean On Me." The sequence alternates directions, going from left to right and then right to left. Interestingly, this pattern illustrates the structure of the melody, which ascends and then descends four notes of a scale. The other example uses a rolling pin to trigger notes on a pattern made of foil, in what appears to be a more aleatoric (random) composition.
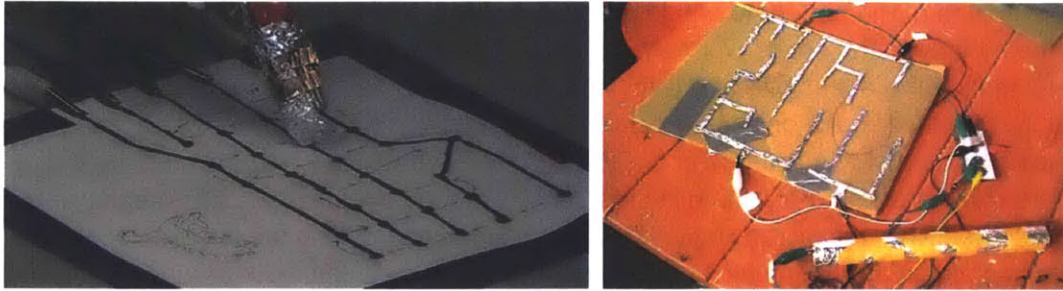
Figure 18: Scores played with a tool. Left: "Lean On Me," with arrows to show the path through the score. Right: score played with a rolling pin.

Connection types

The figure below shows the five possible types of connection that can be used to trigger a MaKey MaKey input. A skin-to-skin connection is triggered when one person connected to ground on the MaKey MaKey touches the skin of a person connected to an input. The human synthesizer in the self-modifying instrument, below, is one example. A skin-to-object connection replaces one of the people with an object having at least a small amount of electrical conductivity. One example is the banana piano shown above. A tool-to-object connection replaces both ends with objects, with one of them being a wand or a mallet that is used to touch the other. The drum kits played with foil mallets work this way. Bridge connections use two separate conductive areas that are connected to each other by touching both of them simultaneously. The floor piano shown above works this way, with each pad made up of two regions that can be bridged by a single foot. Finally, a switch mechanically brings the two connections together. The cardboard bird shown below uses a switch mechanism for the wing, which triggers a sound when the wing touches the body of the bird.
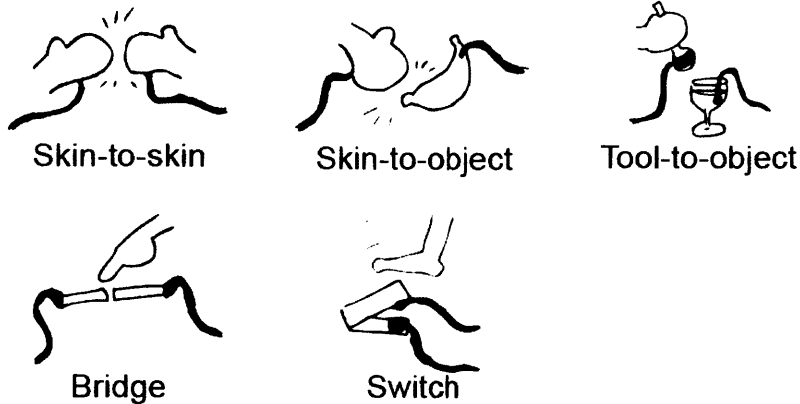
Figure 19: The five types of MaKey MaKey connections.

Workshop Case Studies

I have run about 25 workshops with MaKey MaKey, in a variety of contexts. I've worked with a wide range of age groups, including elementary school, middle school, high school, undergraduate, graduate, educators, and other adult novices and professionals. I've run workshops in the US, Holland, England, Japan, and Turkey. The workshops have ranged from less than hour to tens of hours over several sessions, and from just a handful of participants to a hundred or more. Music has always been at least part of the focus of these workshops, and often the central focus. Here I will share case studies of individual projects from two workshops, one with middle school students and the other with adult educators. Then I will present a third, longer case study of a five-day workshop with undergraduate students, with descriptions of several projects.

Self-Modifying Instrument

This story takes place in a public middle school, where I was assisting a team of three undergraduate music education students to run weekly sessions on making music with MaKey MaKey. This school, like the one I worked with for the MelodyMorph case studies, is a Title 1 school, meaning that it receives supplemental funding from a federal program to assist students who are in poverty or otherwise at-risk. The sessions were approximately 45 minutes long, and we worked with a total of about twelve students in groups of three or four. Each group worked at a table with one laptop with a MaKey MaKey.

In our fourth session with one of the groups, we had set up the laptops to trigger a set of notes in a pentatonic scale, using the Soundplant software. Most of the groups were experimenting with a set of conductive objects and materials (such as play-doh, aluminum foil, and foil baking pans), and using these to play simple melodies. One group was instead making a "human synthesizer," triggering notes using skin-to-skin contact. In the terms of the catalog above, this project is an invention, with a melody-specific pitch arrangement, and

skin-to-skin connections. They used it for an experiment that can be described as a self-modifying musical instrument.

Typically, the human synthesizer instrument can play only a fixed set of sounds or pitches. One person connects their body to ground on the MaKey MaKey circuit board by holding an alligator clip connected to it. Several other people connect themselves to MaKey MaKey inputs for different keys. When the ground person makes skin contact with one of the input people, often by touching their hand, that person's note is triggered.

In the figures below, you can see the variation on this human synthesizer concept invented by the students. There were four students at the table. At the top left of the figure is the student connected to ground. At the bottom left (partly hidden behind a laptop screen), is the student connected to the note C. At the top right is the student connected to the note D. At the bottom right, the student has two inputs in front of him, for the notes F and G. He can choose which note his body will trigger by touching one of two alligator clips, one with a red wire and the other with a blue wire. The "ground" student played a steady cyclic pattern on the hands of the other three students, touching the hand of the F/G student, then D, then C, then D again. As he played through this pattern (shown repeated four times in the music notation in the figure), the F/G student alternated between F and G, changing the melody each time through the pattern. He also experimented, sometimes holding one of the notes through two cycles of the pattern, or holding them both so that the F and G were triggered simultaneously.
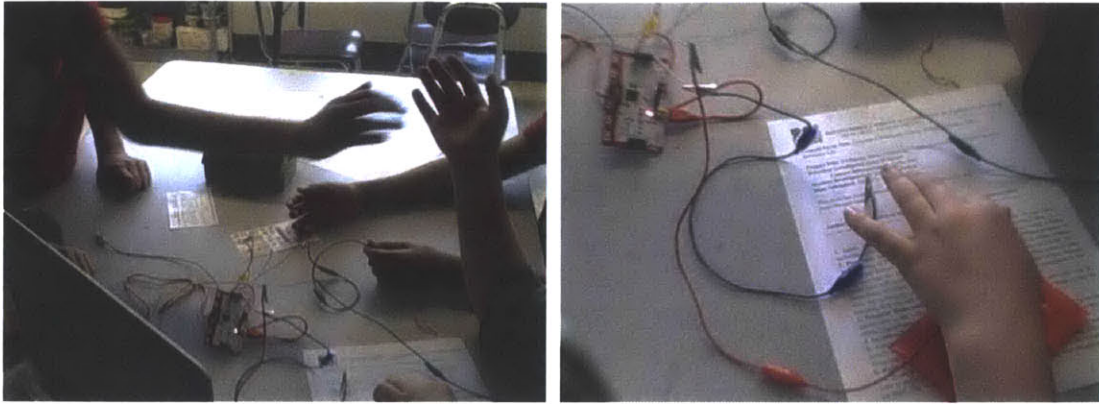
Figure 20: Self-modifying human synthesizer. Top left image: the "ground" student at top left, the "C" student, at top right, the "D" student, at bottom right the "F/G" student. Top right image: The "F/G" student's hand, with the blue and red wires for connecting to the notes F and G. Bottom: Standard notation showing four times through the cycle, with the first note alternating between F and G.

As the students played this pattern, I asked "can you guys explain how you did that?" The F/G student replied "just, he's doing the thing, and I'm changing the note." His casual answer suggests that this seemed to him to be a simple and easily understood experiment, but it was an invention I had never seen before in a workshop, even with musically sophisticated adults.

This setup is made complex, first of all, by of the number of connected parts: the table of students mapping bodies to clips; connections on the MaKey MaKey mapping clips to keyboard keys; and a computer with software mapping keys to pitches (see diagram in the figure below). This set of many parallel connections is a complex situation to grasp. In order to tinker with such a setup, it is easiest to take most of it for granted, as fixed, and modify only one small part at a time. There are several options for how to change it: to change the set of pitches, you can change the pitch of the sound in the software, the relationship between the key and the sound in the software, the arrangement of cables clipped to the MaKey MaKey, or the choice of cables connected to each person.

Person   Clip   Key   Pitch

A ⟶ yellow → w ⟶ C

B ⟶ green → a ⟶ D

        red → s ⟶ F

C ⟨

        blue → d ⟶ G
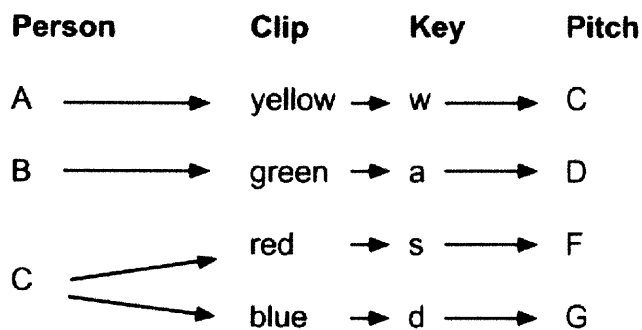
Figure 21: Connections in the self-modifying human synthesizer. Each person holds an alligator clip, which triggers a computer key, which plays a note of a particular pitch. Person C can choose either of two clips, in order to play two different pitches.

In this case, the "toggling" student is changing which cable he is connected to, and doing so in real-time as the "ground" student performs the rhythmic pattern of taps on each of the other students' hands. The making of this change in musical time, as opposed to outside of a performance, illustrates a decoupling of the properties of pitch and rhythm. The ground student is performing the same rhythmic pattern each time, but the moves made by the F/G student change the resulting set of pitches. Typically, musical performance consists of a set of continuous musical events that bind together these properties: there are no pitches without a rhythm, and no rhythms (in a melody) without a pitch. The ability to manipulate the performance by decoupling them shows an abstract understanding of these properties in action.

The tinkering map, in this case, is made up of the students' set of hands around the table. The ground student plays on a path through this map, triggering notes by touching the hands in a pattern. The F/G student manipulates the map by changing his connection. The combined properties of symbolizing and acting in a tinkering map are here embedded in the students themselves, or in their hands. For the ground student, triggering the notes by touching hands, each of the hands refers to a note or to a pair of possible notes. One interesting aspect of the tinkering map in this case is that the F/G student's hand becomes a kind of indeterminate or unstable symbol. This creates a special unpredictable quality for the ground student playing the instrument. Unlike other musical systems with some degree of physical or computational randomness, in this one the unpredictability is caused by the agency of another person.

This unpredictability makes for a curious type of tinkering loop. The ground student, as he performs, is having a kind of conversation with musical materials, but some of those materials are being controlled in real-time by another person. So we see a superimposing of two kinds of loops. "Backtalk" from an unexpected musical event typically arises from a physical or musical property, but in this case is caused in part by a person modifying the instrument. And

the interpersonal feedback in musical tinkering is in this case mediated by the behavior of the instrument itself.

Train of Thought

This case study comes from a workshop I organized along with other members of the Lifelong Kindergarten group for the Creative Learning Network, a gathering of local educators. The participants came from a variety of backgrounds, but all shared an interest in working with youth in creative ways with technology. Many of them already had some experience with MaKey MaKey and with Scratch programming, and some had expressed a desire to learn ways to go deeper with their MaKey MaKey projects. I had also been thinking about ways of helping people in general to find more open-ended possibilities with their MaKey MaKey music creations. I had been hoping to find ways to help people get past their initial simple projects and into more complex creations. I realized that in the past I had found myself doing this by asking questions of workshop participants like "How could you make your instrument much larger? Or much smaller? Or easier to play, or harder? Could you wear it on your body, attach it to the chair or table, or make it so big it has to be played by several people at once?" A full set of questions that I wrote up for the workshop is shown in the table below. Because a very common first simple MaKey MaKey project is the banana piano, I called the workshop "Beyond the Banana Piano."

Instrument metaphors
Can you make something that works like a:
trumpet? trombone? clarinet? guitar?

Non-instruments!
Can you make a musical instrument that works like a:
filing cabinet? spray bottle? toothbrush? stapler? book? chair?

Playable scores
Can you make a drawing that plays music when you touch it?
now can you make it simpler? more complex?
more like standard notation? less like it?
much larger? much smaller?
a three-dimensional musical sculpture?

Other sounds
You can play individual notes, but what about percussion sounds, mouth noises, crazy sound effects, words, musical phrases, chords?

The body
Can you make it so big it has to be played by several people at once?
So small you can barely see it? What if it's arranged so that you have to move your whole body to play it, jumping or spinning? What about an instrument made of bodies?

A choreographed dance piece where the music is created by the dancers touching?

Materials
What else can you make an instrument out of?
Water? Food? Furniture?
Cardboard? Metal? String?
Coffee cups? LEGO bricks? Plastic storage bins?

Table 2: Questions designed to help workshop participants reach more open-ended possibilities with their MaKey MaKey music projects.

One of the groups, perhaps inspired by these questions to combine elements in their project in a new way, made a musical train that played the first phrase of "Mary Had a Little Lamb." By my categorization scheme above, this project is in the automation family, uses a score-like pitch arrangement, and makes connections using switches. The construction of the musical track for their train revealed something about the participants' musical thinking. The group used a LEGO WeDo kit to control a motor that moved a little LEGO car across the table. The car had an arm sticking out from the side carrying a strip of copper tape connected to ground on the MaKey MaKey. The car would drive past a row of five lumps of play-doh, each connected to different keys on the MaKey MaKey. Each lump of play-doh had a paper cut-out stuck in it showing the corresponding syllable in the song: "ma," "ry," "had," "a," and the last one with three connected lumps combining three syllables: "lit," "tle," and a drawing of a lamb.
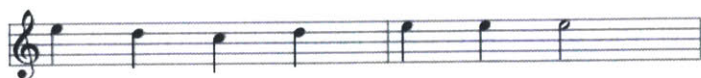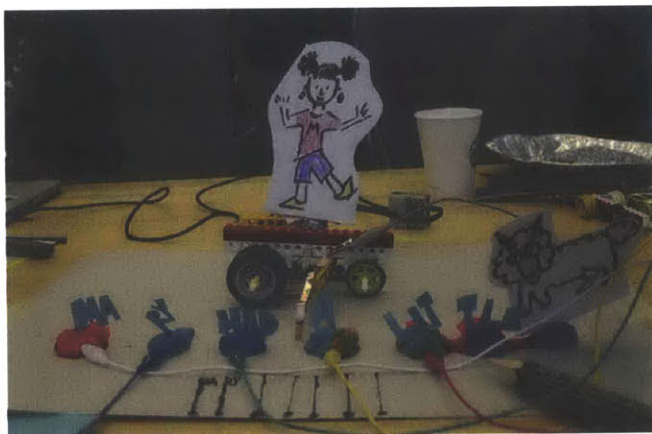
Figure 22: Top: A musical train that plays "Mary Had a Little Lamb" as it moves along. Bottom: Notation for the melody that it plays, approximately lined up with the photo.

At first glance, the project appears straightforward: each lump of play-doh triggers a note in the melody. But closer inspection reveals some peculiar features. This particular folk melody is quite simple, containing, in this first phrase, only three different pitches. So first, we might ask, why does the project need five separate connections, instead of three, one for each pitch? Some of the connections are redundant, triggering the same pitch. The last three syllables, for "little lamb," are all played with a single lump of play-doh that is made from three connected lumps, so that the foil arm, dragging past it, will trigger that one note, an E, three times. In this case the participants saw they could use the same connection to trigger the same pitch. But the first note in the melody, for "ma," is also an E, so why does it need its own connection? The second and fourth notes of the melody are also on the same pitch, a D. Why do they each have their own connection?

These curious features reflect the role of the sequence of lumps as a tinkering map, revealing something about the musical thinking of its creators. Part of the reason for the redundancy in the track may lie in the sequential process of constructing a musical train, and part in what Jeanne Bamberger calls "structural function" (Bamberger and Hernandez, 1999). The assembly of the track proceeded, we can imagine, in a linear form, progressing from one pitch to the next, much as the vehicle proceeds. With this linear, path-like view of the melody, one sees the note ahead only in terms of its difference from the preceding one. This relative view of pitch, rather than a view that situates pitches on a fixed, absolute scale, might make it hard to see the fact that two notes that are not adjacent in time or in space have the same pitch. Indeed, the three notes at the end that have the same pitch and are adjacent do share a single connection. It's only where notes are separated that they have separate connections.

Bamberger's idea of "structural function" (1999) is the idea that a note's placement in the structure of a tune determines the function it plays, which influences how we hear it. In this case, the E at the beginning, and the three E's at the end, have different structural functions. The E at the beginning is a starting point for the melody, and the ones at the end create a resolution. Even though when we isolate their pitch property these notes are all identical, their different roles give make them different meanings. This difference in meaning can make it very hard to separate out the pitch from the context and hear them as the same.

We can see evidence of the group's progress through a tinkering loop in the remnant of an earlier iteration of the project. At the bottom of the photo, you can see seven parallel lines of conductive paint. At the bottom, each has a circle for an alligator clip to connect to it. At the top, each line has the corresponding syllable written out. This earlier version of the tinkering map has a separate connection for each syllable in the melody. The play-doh version goes goes from seven down to five connections, by taking advantage of the repeated pitch of the last three notes. This difference may reflect the difference in working with these materials. Conductive paint takes several minutes to dry, so it can not be easily tested during the painting process, and modifying it after testing requires waiting for it to dry again. Play-doh, on the other hand, is easily reshaped while it is connected to the circuit, allowing it to be continuously tested and modified. Perhaps the play-doh version also started out with seven

separate connections, but as the creators were setting it up they realized that the last three notes were the same, and connected them to each other. The play-doh connecting these three lumps is a distinct piece of a different color, suggesting that it was added later.

If this group had been able to progress through additional iterations of their design, their tinkering map might have helped them make further progress, both in reducing the redundancy of their design, and in understanding the properties of this seemingly simple melody. In addition to helping them simplify their system by reducing the number of inputs, this process might have helped them identify the same pitch in different locations, isolating each note's pitch property from its structural function.

Music as Material

I was invited to collaborate with the ArtEz Institute of the arts, a university in Holland, to run an interdisciplinary workshop for a group of about 15 students. They came from a range of fields including music therapy, computer science, theater, industrial design, and fine arts. I invited a Lifelong Kindergarten alumnus, Michael Smith-Welch, to assist in organizing and facilitating the workshop. The workshop ran for five full days. We were given free use of several rooms in a university facility, a wide range of materials in addition to what we brought with us, and an assistant to make quick trips out for additional supplies. The hosts provided meals in a common space, lectures from invited guests on a range of topics, a brief field trip to see local historical sites, and the opportunity to set up a public exhibition of our work on the final evening.

Here is how I described the concept for the workshop in a summary I wrote for our workshop hosts:

> "We arrived at the name 'Music as Material' for the workshop because we would be combining explorations of both musicality and materiality. We could explore musicality in the design of interactive instruments and the sounds they generate. And we'd explore materiality in creating the physical forms of the instruments out of various materials including paper, fabric, metal, balloons, play-doh, and whatever else we find around us. The name is also meant to capture the kind of conversation we hoped to create as the participants found their own ways to relate musical ideas and physical materials to each other, asking questions about how different materials relate to the body as musical instrument interfaces, how the materials themselves influence the music the player makes, and how physical materials can be used to represent, generate and manipulate musical ideas."

The work the students did at this intersection of musical ideas and physical material made for a rich conversation. Looking back I see numerous examples of tinkering loops and tinkering maps in the students' processes and projects. Here I will discuss three of their creations: The musical gloves, The water-spraying instrument, and the balloonophone.

Musical Gloves

A student named Frank modified a pair of latex gloves so that his thumbs were bare, and wrapped foil around each finger tip. By touching a thumb to any finger, he could play a note. In this way he could assign eight different notes to his fingers. In the terms of my catalog, this is an invention, with a piano-like pitch arrangement, and skin-to-object connections. Frank discovered that with the eight notes in a full octave of a major scale on his two gloves, he could play the beginning of "Ode to Joy." Playing around with this melody he discovered that he could play a kind of glissando, a rapid run of notes, by sliding his thumb across several fingers. For example, he could slide up to the first note of the melody, the top C in the scale, by quickly moving his thumb across his pinky, ring, and middle fingers, before landing on the index finger. Another unexpected discovery he made while playing with this system was that he could play chords. Even though he only had the one thumb per hand to trigger notes (instead of the several fingers one uses, for example, to play chords on a piano), he could make chords by pressing together different combinations of fingers. Because the foil wrapped all the way around each finger, as long as at least one was connected to the thumb, they could all be triggered at once.

Figure 23: The musical gloves. The body is grounded, so the thumbs can be used to trigger any of eight notes by touching the foil on the fingers. In the image on the right, he is playing a C major chord (C and E the left hand, G and high C on the right hand).

In addition to the "Ode to Joy" performance, using a church organ sound, Frank also connected his musical gloves to other sounds. One of these was a piano sound which he used to play jazz harmonies (he is a skilled jazz musician) while another participant accompanied him on a miniature MaKey MaKey drum-kit, also built by Frank. He also used the gloves to mute and unmute tracks of an electronic music composition, which resulted in a totally different way of performing with the gloves. He extended this system to control distortion and filter effects using several more alligator clips connected to his shirt.

Another workshop participant, Hank, was inspired by Frank's gloves to make his own, which passed through several iterations. One early version of these gloves used a yellow plastic glove similar to the one Frank used, but it had connections for notes on all five fingers, and he triggered them by touching his other hand, which was bare and grounded. By the end of the workshop Hank had created a more polished system, using rings of transparent tape that he put over each fingertip, with metal foil on the outside of the tape. With the wires running along his arm, this design made the system almost invisible. For the ground connection he tapped his fingers against a metal ladder. At our final exhibition, Hank was able perform a very expressive version of Leonard Cohen's song "Hallelujah." He accompanied his own singing by playing the piano using his glove, each finger triggering a different chord.

Figure 24: Hank's musical glove. With foil attached to tape around each finger, he can trigger different piano chords by touching the ladder.

When Frank presented his own gloves in our exhibition, he explained why the design of his musical gloves was important to him: "I love electronic music, but what I don't like is when people just set up a computer and press play. I wanted to have an instrument that you could actually play live." About a year later, he was able to fulfill this goal in an impressive way, when he was invited to perform on a Dutch television show called "Join the Beat." For this performance he built a more polished version of his gloves, with essentially the same design. He created three sets of the gloves, so that he could perform in a trio on the show (one of the other performers had also been a participant in my workshop). Frank sat in the center, playing drums expressively, throwing his hands up and down as he played on the gloves, while the other two played bass and keyboard sounds on their gloves. They performed an arrangement based on the groove from the Michael Jackson song "Billie Jean." The bass line of the song consists of a steady eighth note pattern that uses only four different pitches, making it playable on one hand with the musical gloves.

Figure 25: The musical gloves on Dutch television. Right: the translation of the full sentence he says is "I invented them during an MIT workshop in Arnhem." Right: the caption translates to "the circuits are closed when I touch a finger with my thumb."

The gloves are tinkering maps because their physical layout reflects musical ideas. They have an unusual kind of manipulability, because the shape of the instrument is formed by the shape of the hand. This direct connection between pitch arrangement and the body allowed for intuitive experimentation and discovery of new techniques. The fingers of the gloves also play a role as both visual symbols for sounds, and as triggers to make those sounds. This feature especially supported Frank's wish to make an instrument that made electronic music performance more visible to an audience. One problem is that the mapping is much more visible to Frank, who knows which pitches or percussion sounds correspond to each finger. A fuller legibility for the audience would require some markings or decoration to make the relationships explicit. But perhaps more importantly to Frank, he is able to move his body expressively while he performs in a way that is compelling to the viewer.

There are examples of tinkering loops at several scales in this case study. Frank's initial experiments with his gloves were a rich conversation with materials. He discovered through this process that he could use his fingers in musical ways he had not anticipated, to play rapid scales and chords. He could only discover these affordances by making the gloves and trying them out, to see what was possible. The different ways of using his fingers to create music were not imaginable until he was able to experiment intuitively, finding out what his hands could do. He entered larger loops at the interpersonal level, through collaborating with other workshop participants, improvising along with them and letting them try out his gloves. Henk's gloves, inspired by Frank's, were of course also part of this feedback loop, and probably gave Frank further confidence in the musical possibilities of his own gloves. In fact, these possibilities were so powerful that he was able to develop an even better version for a performance on television. Perhaps this experience inspired him to rethink his own musical career or his relationship to music technology, making it possible to imagine even greater successes. That would make it an example of double-loop learning. Recently, Frank wrote to me in an email about how the project "opened up various doors" for him, leading to important new experiences:

"Word got around about the gloves at Artez (my school). I did a presentation for our classical music students, showcasing the gloves (and letting them try them themselves). I explained various things that were possible with the makey makey, and - as usual - came up with at least a dozen new ideas while talking... The gloves also got me into a masterclass by Imogen Heap in Berlin, where I got to play a piece with her "musical gloves." What these things can do just blew my mind... The gloves - the [music as material] masterclass I should say - have opened up various doors for me. I consider getting a chance to make an appearance on television and talking to one of my musical heroes in person two of the highlights of my life."

Water-spraying instrument

The water-spraying instrument emerged after we explored the rooms adjacent to our workshop space and found a large unused bathroom. Two of the students, Ruth and Frank, decided to make a musical instrument that would be played by spraying a stream of water from a faucet at different targets, each of which would trigger a different note. This instrument can be categorized as an invention, with a piano-like pitch arrangement, and tool-to-object connections. In the first version, shown in the figure below, they made a row of conductive tape triggers on the tile wall above the sink, each labeled with a pitch: C, D, E, F, G, A, B, C. They attached a rubber hose to the faucet, tied off the end, and punched a hole in it so it could spray a thin stream at the wall. The metal faucet was connected to ground, so that the stream of water, as long as it was continuous and did not break up in the air, could complete a circuit when it hit one of the tape targets and trigger a note.

The students' intention was to make a monophonic instrument (one that plays a single note at a time), but they quickly found that the behavior of the water caused it to play several notes at once. One problem was with the alligator clips, which were chained together so they could reach far enough and taped to the wall. If the water dripped down and touched the uninsulated places where the clips were connected to each other it would trigger an unintended note. After taping these to keep the water out they still found that they were getting multiple notes at once. The water was forming a film on the wall across the tile, bridging several of the foil targets together so that they would all trigger. In order to prevent this problem, they built a second version (also shown in the figure), which brought the targets away from the wall and suspended them on wires stretched across the sink. This version was more reliable, but also rather messy (they had to tape plastic sheets all around the sink to keep water off the floor and away from the electronics).

Tinkering loops are present here as iterations in the physical design of the system. Seeing the unexpected short-circuits between notes due to the flowing of water across the wall, they redesigned the system, yielding new possibilities and new problems.
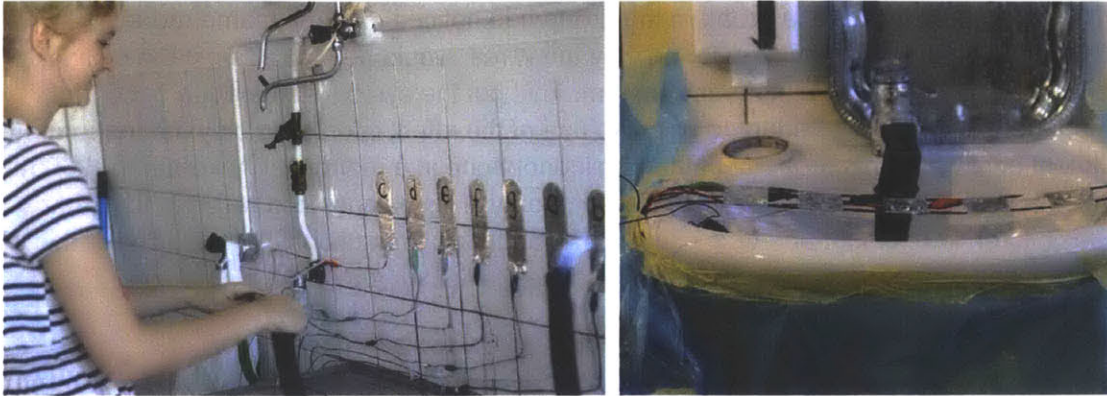
Figure 26: The water-spraying instrument. Left: version one, with targets on the wall. Right: version two, with targets suspended over the sink.

The balloonophone

The balloonophone (also mentioned in an anecdote in the introductory chapter) consisted of a set of helium balloons that triggered notes when they bounced against a sheet of foil on the ceiling. By my categorization, this project is an invention, with what might be called switch connections, or perhaps tool-to-object connections. The pitch arrangement is a bit random, so in this case it does not fit neatly into any of the categories (perhaps it is other-instrument-like, if we make an analogy to instruments like wind chimes).

This project passed through several rapid iterations, partly because the physical properties of the balloons presented some challenges. The latex balloons were not electrically conductive, so in order to make a connection, they needed to added material to the balloons. But in order for the balloons to float up quickly, they needed to remain light. Because aluminum foil was too heavy, the group experimented with conductive paint. Would it adhere to the balloons? Would the paint shrink as it dried, damaging the balloon? After determining that the paint would work, they decided to paint spirals on the tops of the balloons, to make enough surface area to get a good connection when the balloon hit the foil. But now they needed to get a connection from the balloon to the circuit in a lightweight way as well. They painted a line of conductive paint from a spiral at the top down to the knot, and tied a piece of conductive thread to the knot, covering this connection with the paint. But how to connect the thread to the MaKey MaKey board? Alligator clips made the balloon heavy again, and they also quickly discovered that the conductive thread, without insulation to protect it, would cause a "short-circuit" between multiple balloons. If the threads touched, each balloon would trigger all of the notes. Finally, the group settled on running insulated wire to the MaKey MaKey, mounted on the ceiling rather than the floor to make the wires shorter. They also created a connector using perf board to hold the wires in place on the MaKey MaKey, so they wouldn't get pulled out by enthusiastic balloonophone players.

With this full system in place, a problem the group had noticed earlier became more significant: the balloons, made heavier now by the wires, would bounce against the ceiling, triggering repeated notes in a stuttering rhythm. This put the system at the edge of control, requiring the player to either adapt to or make use of the effect in order to play the instrument in an intentional way. This property put the balloonophone in a realm that Ackermann (2005) describes as balancing responsiveness and autonomy. This balance can result in toys that engage children in a dialog, and help explore the feedback interactions they also find within themselves and with others. How does this balance play out in a musical instrument?

Figure 27: Building the ballooonophone. Top left: Tying conductive thread to a balloon with a line of conductive paint on it. Top right: Painting a spiral of conductive paint. Bottom left: a connector to hold wires in place. Bottom Right: installing the balloonophone.

In this case, the students preferred to make an instrument that allowed them more direct control over timing, and less "autonomous" behavior due to the physical properties of the balloon. One member of the team worked on custom software to filter out the additional bounces, so that letting a balloon float up and hit the ceiling would trigger just a single note. In

our critique session they generated numerous ideas for improvements, including the idea of triggering a note not when the balloon hits the ceiling, but when it is pulled down from the ceiling. Also, it was difficult to control several balloons at once. They suggested making a controller with finger holes, so that each finger could control a balloon, or pedals, so they could be controlled with the feet.

There is a rich tinkering loop in the complex iterative design process of the balloonophone. It involved experimenting with the conductivity and buoyancy of the balloons, the affordances of the balloons as tangible controllers, and the interaction of musical properties with each of these. And the balloons as an instrument created a complex landscape of possibilities, a tinkering map which the students worked to shape to their liking.

Reflections on the "Music as Material" workshop

The "Music as Material" workshop was a powerful experience for me, and the fullest expression thus far of my ideas about musical tinkering. The open-ended structure of the workshop was an important aspect, allowing the students' playful experiments and interests to guide us toward their emergent goals. This open-endedness supported what Sawyer calls "collaborative emergence," in which new structures emerge not through top-down control but by bubbling up from many small interactions. This aspect was especially important to the students, since it was a new experience for some of them. Here are some reflections I wrote in an email a few months after the workshop:

> "One of the most exciting parts of the workshop for me was hearing from the students about how totally different their experience with us was from their typical studies at the University. I care very much about the process of learning experiences: giving people a chance to tinker, explore, fail and try again, and explore their own interests. The students really appreciated the freedom to learn in that open-ended way. I wrote down a quote from one of the students, who said the workshop was 'like solving a puzzle together, only you don't know what the puzzle will be.'"

We introduced the idea that the students would have creative freedom from the very beginning of the workshop. We started the first day with a round of music improvisation games that did not use music technology. We began with a round of the game "Floor-Head-Hum," a singing game that I learned from Ben Vigoda's unpublished book "Games for Song." We lay on the floor in a ring, with heads in the center. The only rules are that the group begin singing simultaneously, sing only steady long notes, and only change pitch when taking a breath. The result is often a mix of dissonance and strange but beautiful emergent harmonies, with a feeling of immersion and connection. The game created a sense of shared experience, and established a very open notion of our musical focus. It also immediately invited the generation of ideas for variations on the rules, and new improvisation games. By consensus of the group we started each of the remaining days with these games. The group's invented rules included a game of "Ligeti-Head-Hum," named for the modern composer, with pulsating syllables, and other rules that created gradual convergence toward

or divergence from a single pitch. Of course these games quickly made clear which students were most comfortable with leadership. We encouraged others to try out leadership roles as well, for example in improvised singing games shaped by a conductor. Conducting is a powerful and novel experience for many people, especially for those without other musical experience.

This creative freedom and autonomy transferred to other aspects of the workshop. One student found an upright piano in a nearby room, and decided he would modify it, adding MaKey MaKey connections inside the keyboard mechanism. We never asked for permission, the piano was unharmed, and the result was a successful project (shown above under augmented instruments). The students also of course guided the direction of all their own projects, changing course and adapting when problems arose, and finding and requesting materials as needed. And they played a role in shaping the structure of their final exhibition, choosing how to arrange their interactive installations throughout the space and how to present them.

These reflections suggest to me that the "music as material" workshop supported the growth of complex feedback loops in musical tinkering. There were numerous examples of tinkering loops at the lower levels throughout the case studies. These included conversations with materials that played out in different ways in the design of the musical gloves, the water-spraying instrument, and the balloonophone. For the gloves, thinking with the hands provided useful backtalk. For the water sprayer and the balloonophone, the subtle dynamics of the behavior of water and of balloons both constrained and guided the design. Tinkering maps appeared in these subtle ways throughout the workshop, in the ways that the arrangement of physical materials both reflected and influenced musical ideas.

Loops also played out at the highest level, beyond the time span of the "music as material" workshop. One of the projects, the musical gloves, continued well after the workshop, even finding its way into a performance on television. Is this an example of double-loop learning (Argyris and Schön, 1978)? It may be that the experience of inventing a meaningful music technology, and performing with it on television, helped to transform that student's understanding of what it would be possible for him to create in the future. Another student wrote to me that the workshop had helped him decide to change his path in school, to focus more on creative technology. He wrote that it changed his attitude toward his work and even toward himself: "The 'Music as Material' workshop was life changing, because not long after that I completely altered the route of study… which resulted in some awesome projects. I got really motivated again and became more comfortable in my skin."

Educator Case Studies

These case studies provide glimpses into the work of educators and their students using MaKey MaKey in two different contexts: an urban elementary school and a library makerspace.

Interpersonal beat making

Martin Urbach is a music educator, percussionist, and teaching artist working in public urban elementary and middle schools. I interviewed him to learn about his inspiring work using MaKey MaKey with his students. I conducted the interview in a very informal, exploratory style, asking questions to get at the many aspects of musical tinkering in his work.

Martin's work makes use of tinkering loops across scales, from intuition and iterative design, to autonomy and imagination. Loops involving intuition played a role in an experiment with second graders, making music using water in bowls with MaKey MaKey. With their bodies grounded, they could dip a hand in the water to trigger a note. They would toss water in the bowl to see if it would make a connection, which only sometimes worked. Gradually, more and more sounds were triggering unexpectedly, as everything got wet. A few of the children responded by getting paper towels, and running around to each station to dry things off, "on call" to help the other kids solve problems. Martin thought that most of the kids did not necessarily understand the concept of closing a circuit, and of the water causing unexpected short circuits. But he created a classroom environment where they could just "go for it," without fear of being incorrect, and use their intuition to guide rapid experiments.

His activities also engaged students in longer loops, for example in a project where the students designed a human synthesizer the size of an entire auditorium, so that all of them could be part of the instrument. Students from grades two through five were involved, with the oldest of them taking charge of the process of prototyping and debugging their connections. Even the youngest students were involved in generating ideas, imagining even grander projects like a whole musical house or an entire musical playground. Over the course of two afternoons, the students wired up the entire auditorium with their own musical design.

Martin also leads activities that combine visual art, poetry, and music, with the aim of helping the children to understand their own identities and to believe in themselves. In one of these, for example, a student drew the picture in the figure below, with the words "we are not afraid," and a self-portrait showing himself shouting the words, with arms raised. The picture is also a musical instrument that triggers two percussion sounds when he touches different parts of a dark line. In the activity, the whole group of students sing a version of the civil rights song "We Shall Overcome," accompanied by percussion played on the drawing. The song includes breaks in which the students chant "we are not afraid." We can hope that this activity, and the many other related activities that Martin leads, will help change the students view as they develop of their role not just in music and artistic creation, but in the wider world as well.
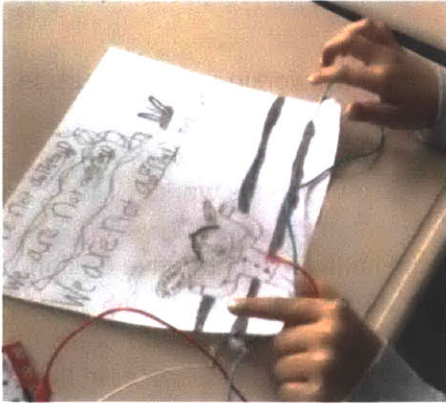
Figure 28: A drawing with the words "we are not afraid" which also functions as a MaKey MaKey percussion instrument.

The tinkering maps in Martin's work are often made up of the children's bodies. For example, he uses embodied activities to help them learn musical duration, which can be a difficult concept. This is especially the case with percussion, because durations are not performed as sustained sound, as on a wind or string instrument. Instead, the durations are made up of silences between discrete sounds. Martin helps his students understand duration by feeling it in their bodies, not just as a "count in the brain." One way he does this is by having the students make different movements when they play rhythms on a MaKey MaKey connection between two people. For shorter notes, like quarter notes, he has them high-five each other. For longer notes, like half notes, they shake hands. The gestures help them experience durations with their bodies, making them into actions they can both feel and see. These transient gestures may not qualify as tinkering maps, because they do not persist in time. But in one activity with middle-school students, he assigns pairs of students each a different duration, which they have to repeat in a steady rhythm. So one pair, for example, is high-fiving a quarter note pattern, while another pair shakes hands on the half notes, once for every two quarter note high-fives. This arrangement assigns a duration to each pair of people, so the students can look around the room and see each duration. Each pair both symbolizes a duration and enacts it as a percussion sound.

The circle song is another activity that creates a visible rhythmic structure from bodies. In this activity the students sing a song while standing in a horseshoe shape. At the ends of the horseshoe, one student is connected to ground, and the other to an input that will play a funny sound. On a steady beat along with the song, the students clasp hands in order around the circle, starting at one end. When the chain reaches the other end, the circuit is completed, triggering the sound. In this case the bodies form a different kind of map, in which the chain of people is a map of time, in discrete units of duration.

Martin sometimes refers to his activities using the title "Interpersonal Beat Making: Opening Minds by Closing Circuits." These words capture the spirit of his work. They hint at the

powerful musical tinkering that he inspires, which he situates in a bigger context. In a recent blog post entitled "A Manifesto for Ethical Music Education," he writes about social justice, saying that we should "teach children to use music as a vehicle to be emissaries of peace and change within their own communities. It is our duty to empower our students to be their generations' Pete Seeger (USA), Mercedes Sosa (Argentina) or Ravi Shankar (India)" (Urbach, 2014).

Iterating instruments

Josh Burker is a teacher in an independent elementary school, working with technology with students from kindergarten through fifth grade. He does a variety of inspiring creative projects with his students including long-term explorations with MaKey MaKey and Scratch. Recently Josh was maker-in-residence at a public library and spent a month working with participants who ranged in age from middle school students to retired people. All of their work was focused on making MaKey MaKey musical instruments. Here is how he describes the activity:

> "As a participant in the workshop you will do all or some of the following:
> 1. Create a prototype of an instrument in cardboard, aluminum foil, copper tape, wire, or other materials.
> 2. Learn to program the instrument using the MaKey MaKey and Scratch
> 3. Build a more durable version of your instrument
> 4. Compose a song
> 5. Figure out a way to write down your song so somebody else could pick up your instrument and play the song.
> 6. Perform a concert."

Josh made a video showing the final session of his residency, which I use here to document the musical tinkering processes he facilitates. The video includes presentations of several MaKey MaKey musical instruments, all built from cardboard. These include Josh's own instruments, and instruments made by children including a cello, a flute, and a bird.

As Josh presented his own musical tinkering experiments, he emphasized the importance of iteration, saying "that's like my big drive in education, is to encourage kids to iterate. Too often when it doesn't work right the first time you throw it down in disgust and you give up and you move on. And all of the participants here never did that at all, and improved as a result." He gave a few anecdotes as examples. These included the story of a paper piano, played by an elderly woman visiting the library, who found that it did not have enough keys to play the tune she wanted to play. So Josh built a new one, with more keys. He also told the story of his cardboard trombone, which went through several iterations as he worked out a way to wire up a cardboard tube inside of another tube so it could play different notes as he slid it back and forth.

153

A cello with a bridge

A middle school student named Ben created a cardboard cello. The first version of the cello had strings made from parallel copper tape strips along a cardboard tube. They each triggered a note when touched with a grounded strip on a cardboard bow. It was hard to play the notes individually, so the next version changed the strings into wires. The wires were elevated above the body of the instrument using a cardboard "bridge" (which has a similar function in a real string instrument). These strings were easier to trigger separately, and now Ben was able to compose a piece of music for his own cello. Josh emphasized that Ben was able to compose at home, even without the MaKey MaKey, because he could simulate playing his cello by pressing keyboard keys. Josh said that Ben was "the first person to get to the next step… he composed a song and came back with sheet music. So he could hand his cello to somebody else, and have them be able to perform the masterpiece that he created."
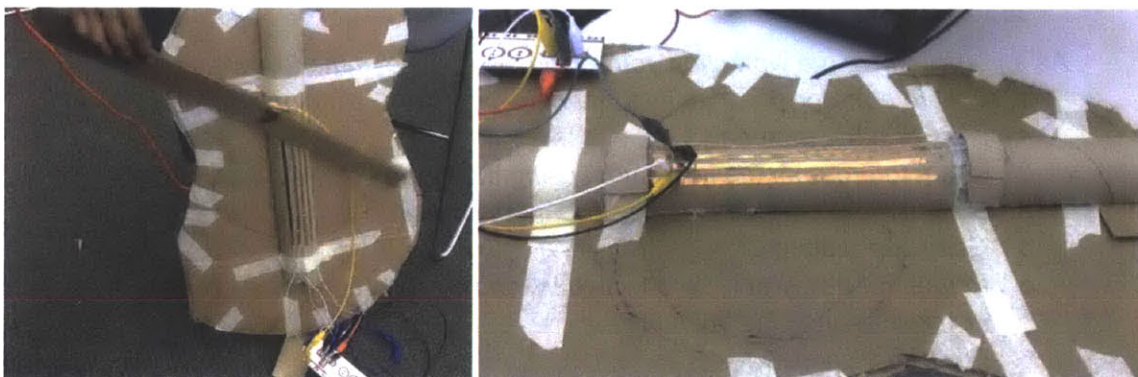
Figure 29: Two iterations of Ben's cello. Left: an earlier version played by touch a grounded bow to strips of conductive tape on a tube. Right: a later version with a bridge that supports wires acting as strings that trigger notes when pressed against the tape strips.

A square flute that can not play rounds

Another middle school-aged student, Sarah, created a cardboard flute. It was a collaboration with Josh, who scored and folded a cardboard tube with a square cross section. Sarah added foil contacts to it to represent the flute keys. But rather than triggering individual notes, her keys each triggered short musical phrases, built as sequences of pitches in a Scratch program. This quickly led to a problem: she didn't like how it was possible to start playing "on top of yourself." At first, Josh and others tried to show her that playing melodies "on top of" each other could be musically interesting. They explained the idea of a musical round by singing "Row, Row, Row Your Boat" (in a round, after the first voice begins, successive voices enter after a delay, creating harmonies).

But Sarah did not want her flute to play rounds. She preferred to change her flute's behavior so that pressing another key would stop the melody currently playing, and start the new one

by itself. Josh helped her use variables in her Scratch program to create this effect. He said, "I was really impressed with her Scratch program. That's a lot of abstract thought for a nine year old, that she's figuring out these complex melodies."
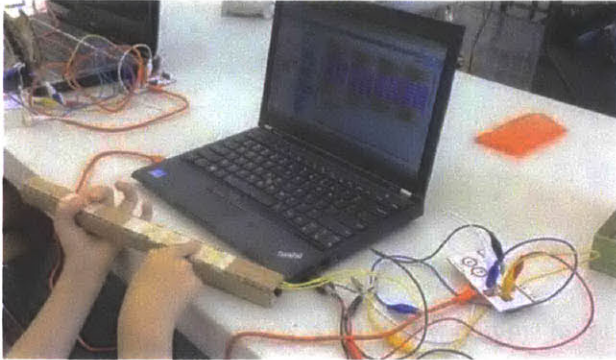
Figure 30: A square flute that can not play rounds. Each foil pad triggers a different short melody, using a Scratch program.

The resulting musical instrument has something in common with Jeanne Bamberger's Impromptu software (Bamberger and Hernandez, 1999). The Impromptu software lets you learn about composing by constructing a sequence of "tuneblocks," which are short melodic fragments each a few notes long. By deconstructing and reconstructing tunes using the tuneblocks, students can learn about ideas like repetition and hierarchical structure in melodies. Perhaps Sarah was interested in a similar type of inquiry, with her flute functioning as a tinkering map. In her map, each symbol both refers to and triggers not just a single note, but something like a tuneblock. But unlike the graphical tuneblocks in the Impromptu software, Sarah's flute keys cannot be rearranged. Her instrument is rigid, but the musical piece she plays on it is flexible. By constructing her own tangible map of a musical landscape, she could explore it in her own way.

A songbird with a human voice

Janna, 11 years old, introduced her project articulately: "I wanted to do something that wasn't a real instrument… I thought a bird would be fun, so I cut one out and then I decided it would be cool to have the beak move, and so now, whenever you open the beak a little bit it sings a low note, then an in-between note when you open it more, and then when it's open fully it sings a high note. And the wing makes a hand clap sound as if the wing's flapping. So I had no idea how I was going to make it, I just thought it would be cool to do. So I cut it out before Josh came and then when he came we figured out how we'd do it in Scratch." She created a three-position switch for the beak on the back of the bird. As you open and close the beak, a piece of foil attached to it touches one of three foil pads that trigger three different arrow keys.
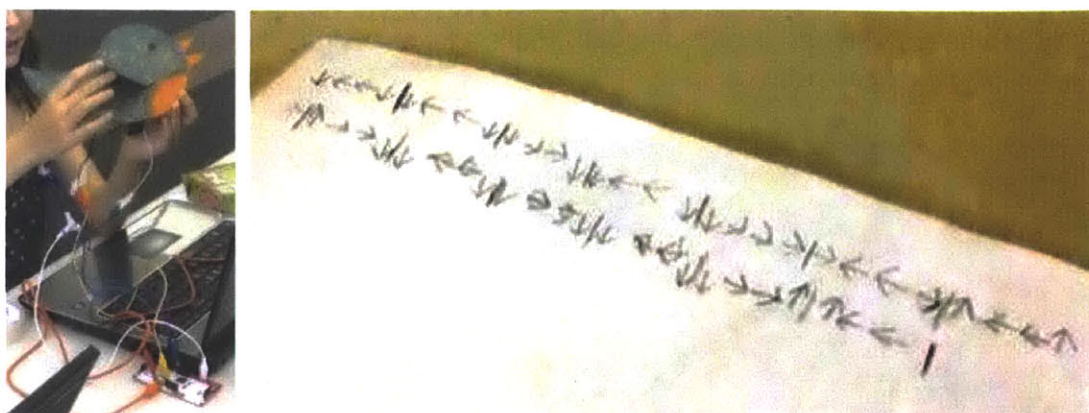
Figure 31: Janna's cardboard bird with a human singing voice. Left: moving the beak triggers three different notes in different positions. Right: Janna's invented notation of her musical composition for the bird.

Figure 32: Janna's bird composition transcribed into standard music notation. The three pitches are played using the beak. Each rest in the notation corresponds to a clap sound made with the bird's wing (a left arrow, in the invented notation above). The final thirteenth bar has only three beats.

Next, she made new sounds for the bird: "Originally I had it making an actual bird sound, but then I decided that it would be too hard because the bird sound had lots of notes in it at one time. And it wasn't really kind of notes, it was more just chirping. So I decided that would be too hard to write a song with so then I changed it to using a human voice. So now the bird sings with a human voice." She used sound editing software to change the pitch of the voice sounds, until she got three that she liked. The pitches of the sounds played by the beak are a low B, a D# in the octave above it, and the high B above that. These correspond to the closed, middle and open beak positions. Flapping the wing triggers a hand clap sound.

Finally, she composed a song to play on the bird, and wrote it out using her own invented notation. Janna's beak mechanism causes the MaKey MaKey to trigger arrow keys, which trigger musical notes in Scratch. She chose to notate these notes using drawings of arrows, rather than a more typical letter to indicate the pitch, or a number, as is used in Scratch. Because she used a sound editor to shift the pitches of the sounds, she may not have known their absolute pitches. Instead, she chose the directional arrows based on the corresponding arrow keys triggered by the beak, which themselves correspond to the position of the beak.

The first few times I watched this video clip, and saw her performance with the bird and her invented notation, I thought that the two were only loosely related, and that perhaps she was improvising. But once I worked out her system and translated the piece into standard notation, I discovered that she had performed her written composition quite precisely. I found that the down, right, and up arrows in her notation correspond to notes played by the beak in the closed, middle, and open positions, and the left arrow indicates flapping the wing to make a clap sound.

Janna's composition has a consistent structure, with groups of four arrows separated by lines like the bar lines in standard music notation. It is read from left to right and top to bottom. The melody shows some compositional sophistication, in its use of a pattern within each bar that both repeats and varies. These features suggest that Janna has had some prior experience with music notation and music composition.

The piece has a suggestion of an "A-B-A" structure, in which an early motif returns after a middle section: bars 2, 3 and 4 are repeated exactly in bars 9, 10 and 11. The piece is twelve bars long, with a partial thirteenth bar for the concluding notes. But the bars in her notation are not laid out in three groupings of four bars each, as might be suggested by the A-B-A structure. Instead it consists of a row of 7 bars and a row of 6 bars. Interestingly, the two A sections (bars 2-4, and bars 9-11) are vertically aligned, appearing in the same position in the first and second rows of her notation. It is worth noting that this alignment is not present in my transcription in standard notation. In this way the invented notation makes the structure of the piece more visible. But it seems possible that the structure actually developed the other way around. Rather than lining up the two phrases on the two rows to show that they are the same, perhaps Janna came up with the idea to repeat the phrase by copying it down from the first row onto the second.

We see evidence of a loop, a conversation with materials, in the form of the cardboard bird. The bird acts a tinkering map, reflecting this musical design process. The bird shows an explicit connection between its form and its sound, both looking like and (to some degree) sounding like a bird. It also shows a connection between its function and its sound, with the mechanical movement of the beak corresponding to the pitches of the notes, and the wing flap making a clap sound. We can see that Janna's desire to make a bird instrument led her to invent the beak mechanism, which in turn both inspired her choice of sounds, and constrained them to three notes. Giving the bird both a beak and a wing mechanism led her to create an

unusual combination: a single instrument that can play two different kinds of sound at once, melodic and percussive. This combination informed the structure of her composition, which uses the sounds in alternation.

Janna's invented notation also acts as a tinkering map, revealing traces of her composition process. The arrows in her notation simultaneously refer to the arrow keys, the direction of the beak, and the relative pitches that they correspond to. The notation has a formal or metric quality, using a consistently measured representation of rhythm, rather than clustered figure groupings. This puts Janna's notation in one of the most sophisticated categories described by Upitis, in the transition between invented and standard forms. But it preserves some evidence of her composition process, in the apparent copying of a three bar phrase from the top row to the bottom.

Reflections

Here I review my observations of musical tinkering in MaKey MaKey. I look at different forms of musical backtalk from physical materials. I examine the relationship of generativity and constraints in musical tinkering, and the role that tinkering maps can play in both. I show how the linking property of tinkering maps helps learners decouple musical properties. And finally I show that in some cases musical tinkering can play a transformative role in a learner's life.

Materials Talk Back

In a conversation with musical materials, sometimes the materials talk back. This unexpected feedback in the tinkering loop is what Schön calls "backtalk." Tinkering with MaKey MaKey can involve a wide range of physical materials, and these talk back in different ways. One kind of backtalk is structural or mechanical. When Frank was creating his musical gloves, he did not expect that he would be able to use his fingers to play chords and rapid sliding glissandos. These possibilities arose as he experimented, and noticed how the intuitive motions he made with his fingers led to new musical ideas. When Janna was designing her bird instrument, she probably did not plan to write a musical composition for it that she would play by flapping the wing and opening and closing the beak. Instead, these interactions emerged from her experiments with cardboard, building the hinged wing and the rotary switch mechanism for the beak.

Another kind of backtalk from materials comes when they behave autonomously. Ackermann (2005) shows how the most engaging toys are ones that balance responsiveness with behavior at the edge of our control. Sometimes tinkering with physical materials means engaging with this kind of behavior. We interact with the materials, shaping them but not entirely controlling them. The water-sprayer and balloonophone projects both involved tinkering with physical materials that form dynamic systems. The spray of water did not simply act like a beam that you point at a target to play a note. Instead, the behavior of the water was messy, dripping and splashing. It unexpectedly formed a film on the wall that short-circuited the connections, requiring a new design. The balloons in the balloonophone also did not quite behave the way the creators wished, because it was difficult to use a balloon to trigger a

single note. Instead, the balloons bounced against the ceiling, making stuttering rhythms. In another case, the instrument was made of people, which are also a dynamic and unpredictable material to make things with. The self-modifying instrument was a human synthesizer, in which three people's hands became triggers for different notes. One of these people behaved unexpectedly, changing his connection so he would trigger a different note.

How does this backtalk, from structures and mechanisms, or from autonomous behaviors, affect the tinkering loop? Unexpected feedback from the material helps tinkerers to generate new ideas. But generating ideas is not enough.

Generativity and Constraints

In addition to generativity, tinkering processes also rely on constraints. Generative processes, like the ones we see in conversations with musical material, flourish most when coupled with a system of rules that limit the space of possibilities. Stravinsky (1959) wrote about the importance of constraints in his own process of musical composition:

> *"...whatever constantly gives way to pressure, constantly renders movement impossible. My freedom thus consists in my moving about within the narrow frame that I have assigned myself for each one of my undertakings... The more constraints one imposes, the more one frees one's self of the chains that shackle the spirit."*

The creative process itself involves selecting these constraints. There is a feedback loop between the processes of generating ideas and constraining them. Sawyer (2002) describes this interaction in his analysis of creative processes in theater and jazz improvisation groups. He sees creativity in those contexts arising from the combination of collaborative emergence and downward causality. Collaborative emergence is analogous to the generative conversation with materials I described earlier. This process involves ideas bubbling up from many small interactions. Downward causality is driven by constraints that come from above. What is "up above" in this sense is the emerging system that is being constructed. In his view, the appearance of these constraints actually helps the improvisers make moves that better contribute to the emerging collaboration.

We see a similar feedback loop between generativity and constraints in musical tinkering. As the self-modifying instrument was being constructed, for example, the students were building a human synthesizer that was constrained by the available parts. They had their own four bodies available, and with one "ground" body, that left three to trigger different notes. With this constraint in place, the ground student could then go about improvising a pattern to play on his instrument of three notes. One of the students noticed that there were additional wires available, and came up with the trick of alternating his connection, so that he could trigger different notes each time. In other situations where people construct human synthesizers, they often simply match the number of sounds to the number of people, and this idea of alternating does not emerge. If there had been more students at the table, or fewer wires, they might not have come up with this new idea.

The musical train example also illustrates the interaction between generativity and constraints. The group limited their project to a small part of the song "Mary Had a Little Lamb," probably so that they would have enough connections to be able to play each note. But even the seven notes in the first phrase require a lot of wiring. This physical constraint helped motivate them to simplify their system by combining connections to the same note. In turn, this simplification may have helped them understand the musical structure they were reconstructing, because they could see which notes shared the same pitch.

The properties of tinkering maps help them support this feedback between generativity and constraints. As ideas emerge, the tinkering map makes them visible, which helps to make constraints visible as well. In the example of the train, the lumps of play-doh illustrate both the musical structure and the number of notes. And because tinkering maps link together symbols and their output, you can manipulate them at multiple levels at once. Physically manipulating the lumps of play-doh is both a way to manipulate them as note symbols, and to trigger their note sounds.

Linking and Decoupling

Tinkering maps can help people learn to understand abstract musical properties. Sophisticated musical thinking involves understanding properties like pitch and rhythm in isolation from each other. The linking property of tinkering maps help people to decouple these properties, by making them visible and manipulable. The human synthesizer was played by one student on a steady rhythm, freeing up another student to manipulate the pitch property independent of rhythm. The lumps of play-doh in the musical train helped its creators separate out the pitch property, shared by the last three notes, from the distinct structural function of each note in its context.

Transformative Loops

Musical tinkering has the potential to be transformative. We saw evidence of the largest tinkering loop, double-loop learning, in a few different ways in the "Music as Material" workshop. The structure of the workshop emphasized autonomy for the students, beginning with playful music improvisation games, and extending to their role in guiding their own projects and in shaping our final exhibition. The student who modified an acoustic piano without permission in advance certainly asserted this autonomy. And ultimately the effects of the workshop extended beyond it. Frank's musical gloves opened up new opportunities for him including getting him on television and into a workshop with one of his musical heroes. More importantly, it contributed to his expanded sense of his own capabilities. Another one of the students wrote that our workshop had changed his course of study, but also deepened his motivation and made him feel more comfortable with his own creative path.

Martin's work is also transformative for his students. He connects music education with social justice, and changes the way his students see their role in the world. The self-portrait projects he does with his students combine visual art, poetry, music and technology. These become

tinkering maps that include not just pitches and melodies, but also emotions, stories, and identities. Creating an interactive self-portrait becomes a way to tinker with who you are, and who you might become.

What is it about MaKey MaKey, and musical tinkering in general, that makes these transformative experiences possible? Perhaps the power to transform everyday materials, constructing your own musical invention, helps people imagine that they could make even bigger changes. The powerful feedback loops driving the creative process may serve as an example for how other creative endeavors in their lives might work. Of course it is worth emphasizing that musical tinkering with MaKey MaKey is not sufficient by itself to bring about these transformative learning experiences. In each case, a supportive learning environment, and especially a relationship with a caring teacher or mentor, plays a central role. But musical tinkering is the catalyst.

Chapter 6:
Conclusion

In this concluding chapter I first present some ideas that synthesize my observations of loops and maps. I translate these ideas into concepts for designing for musical tinkerability. Finally, I imagine future possibilities for musical tinkering.

A Synthesis: Loops and Maps

Here I present four ideas that synthesize my observations about loops and maps across the case studies: the many languages of backtalk, stimulating emergence, new maps for new instruments, and tinkering with ideas and attitudes.

The Many Languages of Backtalk

Tinkering loops often involve a conversation with musical materials. Sometimes that material does something unexpected, and the surprise leads to a new insight. Schön calls this phenomenon "backtalk." Reviewing the case studies of tinkering with MelodyMorph and MaKey MaKey, I see that backtalk can take several forms, depending on the nature of the tinkering map involved. Backtalk can speak through memory, movement, mechanism, and material.

Memory backtalk occurs when a random scrap of melody calls up a musical memory, leading to deeper exploration. We see it in Sandy's early explorations of MelodyMorph.

Movement backtalk occurs when a body gesture, such as a MelodyMorph path, generates unexpected musical output that can be investigated through further gestures. We see these gestures in the evolution of the looping instrument and in the musical gloves.

Mechanism backtalk comes when a physical mechanism being constructed, such as a hinge or switch, produces an unintended musical result that leads to a new idea. This backtalk appears in the making of the beak of the cardboard bird. The self-modifying human synthesizer is a special case, in which the mechanism is made of people.

Material backtalk arises when the physical properties of materials, such as water, paper, or play-doh, give them an unexpected musical behavior. We see material backtalk in the water-spraying instrument and the musical balloons.

Backtalk propels tinkering loops forward. Noticing the many forms it can take can inform the design of future platforms for musical tinkering. Perhaps we can design them to talk back. There may be many more languages for them to speak.

Stimulating Emergence

Tinkering maps help to create feedback between generativity and constraints. This interaction can stimulate the emergent blossoming of creative ideas. Tinkering maps can accelerate this emergence because they make musical thinking visible. Seeing musical ideas as they are generated also makes visible the constraints they create, which can lead to new ideas. We see this loop in the cases of the self-modifying human synthesizer and the musical train with its play-doh score.

The manipulability of tinkering maps plays an especially important role in the feedback between generativity and constraints. If the musical ideas were visible but displayed in a static and unchangeable way, there would be no seamless way to experiment and explore variations. We see the students using this manipulability in MelodyMorph as they edit their creations, rearranging pitches and re-making paths and recordings. The importance of manipulability is also shown in the musical train example by the difference between conductive paint and play-doh. After the group switched from paint to play-doh they could modify their creation more quickly and easily add new connections.

The linking property of tinkering maps is also critical for stimulating creative emergence. By directly connecting visual symbols to musical sounds, tinkering maps allow the feedback between generativity and constraints to proceed rapidly. With only a minimal delay between making and listening, tinkerers can enter a creative flow of experimentation. And the co-location of symbol and action enables learners to tinker with them simultaneously. For example, the musical control of a looping path in MelodyMorph, and the visual symbol depicting the path, exist as one object on the screen. The cardboard beak of the bird is at once a visual symbol of the three notes it can play, and the controller for playing those notes. Tinkering with one is also tinkering with the other.

New Maps for New Instruments

In the introduction I raised this question: when freed from the constraints of acoustic musical instruments, can people invent totally new kinds of instruments?

People often begin with MelodyMorph by building a simple scale. This is the strategy I call anchoring, which may be common in tinkering processes. In this case, anchoring supports exploration of a new tool, by first building something based on a familiar pattern. I see an analogy to Piaget's (1953) concepts of learning through assimilation and accommodation. Assimilation is the incorporation of new ideas by fitting them into our existing mental structures. Accommodation is a deeper change, which becomes necessary when integrating a new idea requires rearranging the existing structures. In musical tinkering, we see people using the tools at first to create arrangements that match or derive from the existing systems that they are familiar with. They make pitch arrangements based on familiar musical instruments or music notation. Only once they build up fluency with the tools are people able reframe the system, and invent radically different structures.

Some people do create patterns of interaction using MelodyMorph that go beyond standard metaphors from acoustic musical instruments. The creation of musical landscapes to support musical navigation is one approach I have observed that is distinct from the way musical instruments are typically used. Melody-ordering, the arrangement of a sequence of bells in the order of a melody, is another approach that goes beyond the possibilities of acoustic instruments. In effect this creates a musical instrument specific to a particular melody. Acoustic musical instruments are not normally re-arrangeable in this way, and so instead are designed for more general-purpose use. The ability to embed recordings and paths into MelodyMorph amplifies this power to make instruments specific to a particular musical composition. One example is the looper instrument created by the students. These instruments can have recorded content embedded in them that specifically supports playing the melodies, harmonies, and accompaniment of that particular composition or musical context. We might call these "composition-instruments."

Another way in which MelodyMorph creations go beyond traditional musical instrument structures is in the creation of musical cartoons and musical stories. Musical compositions have long been associated with imagery and narrative. But MelodyMorph supports a uniquely close coupling, in which a drawing can contain musical content embedded within it. We might call these "story-instruments."

MaKey MaKey instruments also often begin by imitating familiar instruments and familiar pitch arrangements. As in MelodyMorph, they often start by building a scale, which becomes a piano. This anchoring is perhaps an even stronger effect in MaKey MaKey projects than in MelodyMorph. It may be that people focus their creative efforts on tinkering with materials and connections, instead of tinkering with musical instrument paradigms. MelodyMorph makes it much easier to experiment with ideas like musical landscapes because it lets you create an unlimited number of bells and fluidly rearrange them. MaKey MaKey of course has a limited number of connections, and wiring a large number of them quickly becomes complex and can make them hard to rearrange. For novices it can also be difficult to modify the set of pitches and sounds available on the computer. We provide a few simple examples, and many other pre-made applications will work. But detailed customization requires familiarity with either professional music production software, or with programming. Scratch provides a kind of sweet spot, because it is very easy to create a Scratch program that can use a keypress to trigger a recorded sound or a note of any pitch. Of course, Scratch was not designed with only MaKey MaKey in mind, so there are still many details to learn in order to use it in this way.

In spite of these challenges, people do use MaKey MaKey to invent new musical instrument concepts. The MaKey MaKey catalog gives an overview of these. Beyond imitation and augmentation of existing instruments, they include non-musical toys transformed into instruments, wearable instruments, furniture instruments, and instruments that transform activities like juggling and dancing into music. Of course, many of these examples are single instances created by musically sophisticated users. I continue to work on ways to help MaKey

MaKey tinkerers go "beyond the banana piano." I use a diverse set of examples to help people expand the space of possibilities they can imagine. To share these examples, I have collected many of them in the MaKey MaKey web gallery. I have also created two compilation videos, each just a few minutes long, that quickly show a large number of different music projects.

Tinkering With Ideas and Tinkering With Attitudes

We might ask what one learns about music through musical tinkering. I see people tinkering with many ideas, but perhaps more importantly, I see them developing new attitudes about their musical identities.

Throughout the case studies, we see people tinkering with musical structures and systems. As they tinker, they inevitably bump into ideas about music that resemble what one might learn in a more traditional approach to music education. For example, we see people grappling with questions about the structural function of notes with the same pitch in different parts of a melody, and with isolating the effects of pitch and timbre as musical properties. We see them using timbre, pitch height, and tempo to express emotion in their compositions. We see them using melodic structures like repetition, variation, and hierarchical structure. They make creative use of timbres and musical patterns as sound effects, they inventively represent spoken conversations as melodies, and they tell stories through music. They also use ideas about musical interaction design, deciding how best to arrange an instrument to be played with one hand or with two, or by multiple people, in synchrony or in conversation. Tinkering maps help make all these ideas visible, and tinkering loops support the many iterative experiments with them.

Is musical tinkering really a viable way to learn these ideas about music? Seymour Papert (1980) describes a thought experiment in which he describes an idealized world for learning mathematical ideas which he calls "Mathland." He makes an analogy to the way people learn languages. We are not surprised that people naturally learn the French language when they grow up in France, or Chinese when they grow up in China. But our present culture does not speak math. The Mathland he imagines is a world where mathematical ideas are so deeply embedded in the culture, so naturally accessible, and so intertwined with meaningful activities and interactions, that one learns mathematical ideas as seamlessly as a child develops a first language. Papert's Logo environment is a step toward creating Mathland. Perhaps musical tinkering tools are a step toward making a "Musicland," where we are surrounded by opportunities to create and play with music, and learning musical ideas comes naturally.

But beyond tinkering with musical ideas, and perhaps more importantly, we see people tinkering with musical attitudes. This process, which plays out in the largest of the tinkering loops, includes people changing their beliefs about their relationship with music, about their own musical abilities, and about their role in musical culture. Through musical tinkering people can learn that music is theirs to make. There is no right or wrong, no correct or incorrect music. They get to choose what to compose and what to create. The criteria for success

come from their own ideas, which evolve through the tinkering loops of making and sharing. The music comes from their own intuitions, their own stories, and the musical cultures they care about. Through musical tinkering people learn that they can make music. They learn that the music they make is valuable, and that others care about it. They learn that they are composers, performers, and inventors. And they learn that they can learn.

In the case studies, we see some examples of tinkering with musical attitudes in these larger loops. Sandy faces social challenges due to his special needs, but his creative confidence in musical tinkering helps him begin to develop a more powerful social role. Perhaps his joyful tinkering with creative technology and music will be part of a successful path for him in school and beyond. And Frank's musical gloves opened new opportunities for him, leading to a television appearance, an opportunity to work with a musical hero, and perhaps new ideas about his possible future.

Behind these deeper personal transformations, caring mentorship has a crucial role. The ideal environment for musical tinkering includes not just the right setting, tools, materials, and activities, but also people who can shape the experience in a positive way. This includes giving learners attention and respect, inspiring them to new musical explorations, and giving them the freedom and autonomy they needed to make discoveries on their own. Early in my work on MelodyMorph with the music education students, as we began our middle school classroom sessions, I was concerned that we were not creating that kind of environment. But as we progressed, I found that we were starting to cohere as a group of caring mentors. One of the music education students wrote, in a reflection at the end of the semester: "however young or old the students are people and deserve respect. When the environment has mutual respect on boths sides some of the best learning can take place." He also wrote that "the key goal is to inspire the kids... to want to make music in some form when leaving the class."

The role of freedom in mentoring seems paradoxical, because less intervention can be more powerful. One of the music education students visibly shifted his teaching style over the course of our sessions from a lecture-based instructional role to a more open and relaxed approach that gave the students autonomy. In a reflection about what he would do differently if he were to teach these sessions again, he wrote: "I would attempt to do even less. Give guidance but allow the students to create their own activities and projects." He also wrote "I learned that it is just as important to let students work on their own to attempt to solve a problem as it is to teach them how to solve a problem. A student will get more out of the situation and gain more understanding when left to determine the answers on their own."

Similarly, mentorship plays a key role in the shifts in musical attitudes I have seen in MaKey MaKey workshops. For example, the transformations of some of the students in my "music as material" workshop were supported by the attention and encouragement that Michael and I gave throughout. We provided the students with emotional support as well as autonomy. The mentoring presence of Josh and Martin in their MaKey MaKey sessions with children also clearly plays a central role. Josh's attention to detail in listening to and developing the

students' ideas, in helping to construct each student's project, and the support he gives steadily over many weeks is surely responsible for any transformative power his workshops may have. Martin's activities help his young students develop a sense of their own identity through musical tinkering, which requires caring individualized attention. He also inspires his students to engage with social justice issues through music. We can imagine that these activities also have a lasting impact on the students' attitudes about themselves and their role in the larger musical world.

Much work remains, of course, to demonstrate that people can in fact learn these musical ideas reliably through musical tinkering, and that their attitudes can change as well. It is clear that this will require not just tinkering tools, but learning environments for tinkering that include the support of caring and attentive teachers and mentors. But the anecdotal evidence across these case studies gives us at least a proof that it can happen. It also gives me motivation to develop the ideas, tools, and learning environments further, and to share them with the world.

Designing for Musical Tinkerability

What have we learned from this work about designing tools and learning environments to support musical tinkering? First I review the design concepts I used in the chapters on the design of MelodyMorph and of MaKey Makey. Then I review some additional concepts that have emerged from my observations.

In the chapters about the design of each tool, I included a set of six design concepts. Three of these design concepts especially support tinkering loops: immediate feedback, scaffolded complexity, and authentic context. Immediate feedback lets you see and hear the results of your experiments quickly. Scaffolded complexity supports the progression from beginner to expert. Authentic context makes meaningful connections to the user's personal and social worlds.

The other three design concepts support the creation of tinkering maps: flexible space, expressive gesture and linked symbols. Flexible space allows you to express musical ideas using spatial relationships in a variety of ways. Expressive gesture translates intuitive body movement into musical output. Linked symbols connect visual or tangible symbols with actions such as generating musical sound.

Beyond these ideas, some new concepts arise from my synthesis of observations about loops and maps above.

Designing for backtalk seems paradoxical. How can we design a tool that actually surprises people? Each moment of unexpected backtalk is unique. We can not design those in individually, but perhaps we can design a system that is fertile with opportunities. These may include memory, movement, mechanism, and material, or other media.

Can we design to stimulate creative emergence? My discussion above suggests that the properties of tinkering maps are crucial for accelerating this process. By making ideas visible and manipulable, and linked to action, we can accelerate the interaction of generativity and constraints.

How can we encourage the design of new maps, and new instruments? We need to let people to anchor in the familiar before experimenting with the unknown. We need to show people the large space of possibilities, to help them imagine something new. And we need to give them tools flexible enough to incorporate a wide range of musical materials and ideas.

Finally, how can we support tinkering with musical ideas and attitudes? Tools designed for tinkering maps and tinkering loops are a start. But it is the learning environment in which the tools are used that gives musical tinkering the power to transform learners' understanding of music and their beliefs about their role in musical culture.

The Future: Re-Making the Musical Landscape

How can we re-make the larger landscape of musical possibilities? Improving the tools is part of this future work. At the time of writing, I have not yet released MelodyMorph to the public, but I hope to do so. Ideally, it will include a sharing system that will support a community of musical tinkerers. MaKey MaKey is in wide use, but there is room for improvement in its support for musical tinkering. This could include better examples, support materials, and custom software tools at a range of different levels of accessibility and flexibility.

I can imagine new tools for musical tinkering. Some of these are incremental improvements to the tools described here. I could make a version of MelodyMorph with a much simpler user interface, or one more specifically focused on musical cartoons. I could also make a version for more advanced users, with more complex features for music composition and for sound synthesis and MIDI output. I can also imagine a version of MaKey MaKey redesigned to focus on music. This could be designed for accessibility and simplicity, and have a small feature set. Or it could be designed for flexibility, with features like two full octaves of chromatic note inputs, and a combination of discrete and analog sensing.

I can also imagine combining the concepts of MaKey MaKey and MelodyMorph into a new system. My own musical tinkering process has led me into some experiments with this combination that I think could become a powerful musical tinkering experience for others as well. The figure below shows a musical painting that I created in collaboration with Sophia Brueckner. It is an acrylic painting on a wood panel. Touching any of the 80 black conductive paint regions triggers a musical note. You can slide your hands over it to play rapidly cascading melodies and harmonies, in the sounds of a harp and a toy piano.

In order to design the complex spatial pattern of pitches in the painting, I made an interactive prototype using MelodyMorph, which you can see at the bottom left of the figure. I was able to design a musical landscape for the painting, and easily modify it. I iteratively redesigned the

168

arrangement on the screen as Sophia improvised her painting, which is why we called it "Simultaneous Composition #1." Along the way I designed a sensing system for the painting by reprogramming MaKey MaKey circuit boards to use capacitance sensing, so it can be triggering simply by touching it, without an additional ground connection to the body. And instead of keypresses, the reprogrammed boards send MIDI to a sound synthesizer.

I can imagine a construction kit that makes the creation of projects like this musical painting much simpler and more accessible. It would combine a touchscreen-based design tool for creating digital musical paintings with a sensing circuit for making physical ones. A user could map out a set of sounds in space on the screen, and then print out a pattern in conductive ink. Then they could connect the printed pattern to the circuit, making it into a tangible musical landscape.
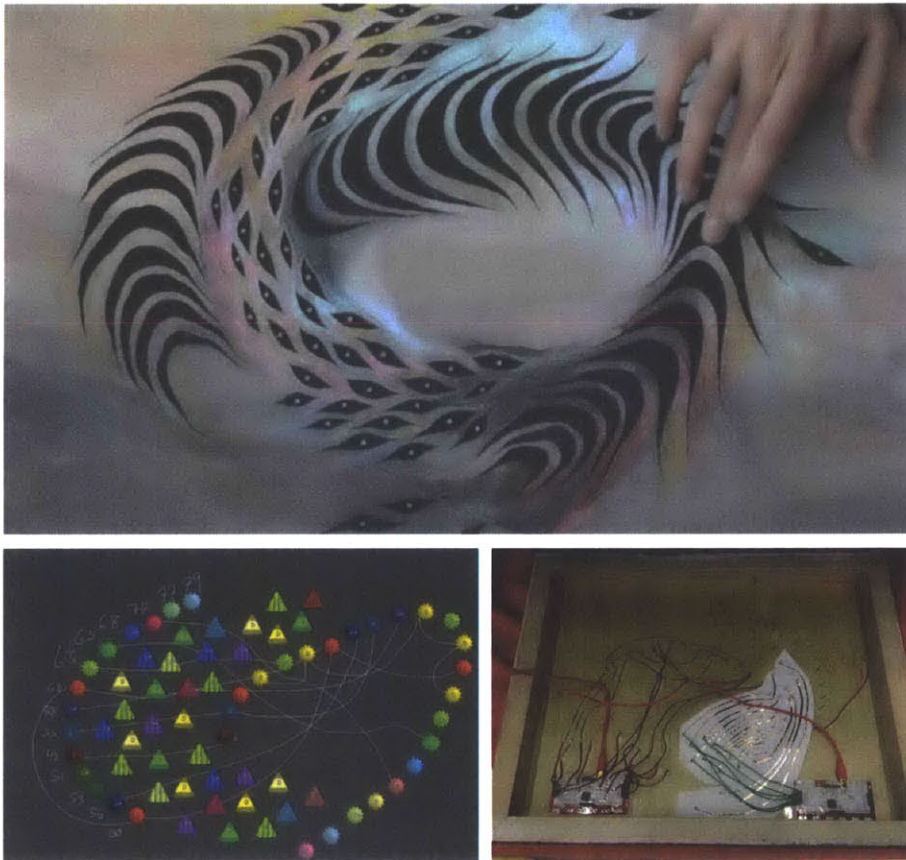
Figure 1: Simultaneous Composition #1, a collaboration with Sophia Brueckner. Top: touching any of the black conductive paint shapes triggers a musical note. Bottom left: my interactive prototype for the painting made using MelodyMorph. Bottom right: the back of the painting, partly wired with two modified MaKey MaKey boards.

The ideas above focus on how we can make musical tinkering tools more accessible, more flexible, and more powerful in new ways. I hope that these new tools will support musical tinkering through musical backtalk, creative emergence, and new maps for new instruments. But I also know that we will need to create learning environments that support tinkering with ideas and with attitudes in order to inspire growth and transformation in learners' musical lives and beyond.

References

Abelson, Harold, Gerald Jay Sussman, and Julie Sussman. *Structure and Interpretation of Computer Programs - 2nd Edition*. Second edition edition. Cambridge, Mass.; New York: The MIT Press, 1996.

Ackermann, Edith K. "Playthings That Do Things: A Young Kid's 'Incredibles'!" In *Proceedings of the 2005 Conference on Interaction Design and Children*, 1–8. IDC '05. New York, NY, USA: ACM, 2005.

Argyris, Chris, and Donald A. Schön. *Organizational Learning: A Theory of Action Perspective*. Addison-Wesley Pub. Co., 1978.

Austin, J. L., J. O. Urmson, and Marina Sbisà. *How to Do Things with Words: Second Edition*. 2 edition. Cambridge, Mass: Harvard University Press, 1975.

Azevedo, Roger, Jennifer G. Cromley, Diane Seibert, and Myriam Tron. "The Role of Co-Regulated Learning during Students' Understanding of Complex Systems with Hypermedia.," April 2003.

Bamberger, Jeanne. *Discovering the Musical Mind: A View of Creativity as Learning*. 1st edition. Oxford ; New York: Oxford University Press, 2013.

Bamberger, Jeanne. *The Mind behind the Musical Ear: How Children Develop Musical Intelligence*. Harvard University Press, 1995.

Bamberger, Jeanne. *Logo Music Projects: Experiments in Musical Perception and Design (A.I. Memo)*. Massachusetts Institute of Technology. Artificial Intelligence Laboratory, 1979.

Bamberger, Jeanne, and Armando Hernandez. *Developing Musical Intuitions: A Project-Based Introduction to Making and Understanding Music Complete Package*. Oxford University Press, USA, 1999.

Berdahl, Edgar, and Wendy Ju. "Satellite CCRMA: A Musical Interaction and Sound Synthesis Platform." In *Proceedings of the International Conference on New Interfaces for Musical Expression*, edited by Alexander Refsum Jensenius, Anders Tveit, Rolf Inge Godøy, and Dan Overholt, 173–78. Oslo, Norway, 2011.

Berman, Greta. "Synesthesia and the Arts." *Leonardo* 32, no. 1 (1999): 15–22.

Blair, Deborah V. "Musical Maps as Narrative Inquiry." *International Journal of Education & the Arts* 8, no. 15 (October 8, 2007): 1–20.

Butler, Deborah L., and Philip H. Winne. "Feedback and Self-Regulated Learning: A Theoretical Synthesis." *Review of Educational Research* 65, no. 3 (September 1, 1995): 245–81.

Collective, Beginner's Mind, and David Shaw. "Makey Makey: Improvising Tangible and Nature-Based User Interfaces." In *Proceedings of the Sixth International Conference on*

Tangible, Embedded and Embodied Interaction, 367–70. TEI '12. New York, NY, USA: ACM, 2012.

Corbin, Juliet M, and Anselm Strauss. "Grounded Theory Research: Procedures, Canons, and Evaluative Criteria." *Qualitative Sociology* 13, no. 1 (1990): 3–21.

Dillon, Steven C. *Music Meaning and Transformation: Meaningful Music Making for Life*. Vol. 1. Cambridge Scholars Publishing, 2007.

Gibson, James J. *The Ecological Approach To Visual Perception*. 1st edition. Boston: Psychology Press, 1986.

Goldberg, Adele. *Smalltalk 80: The Language*. 1 edition. Reading, Mass: Addison-Wesley Professional, 1989.

Gorton, Tim, and Bakhtiar Mikhak. "A Tangible Architecture for Creating Modular, Subsumption-Based Robot Control Systems." In *CHI '04 Extended Abstracts on Human Factors in Computing Systems*, 1469–72. CHI EA '04. New York, NY, USA: ACM, 2004.

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

Hofstadter, Douglas R. *Gödel, Escher, Bach: An Eternal Golden Braid*. Fifth Printing edition. New York: Basic Books, 1979.

Korzybski, Alfred. *Science and Sanity: An Introduction to Non-Aristotelian Systems and General Semantics*. Lakeville, Conn.: Institute of General Semantics, 1933.

Machover, Tod. "Hyperinstruments - A Progress Report 1987 - 1991." *MIT Media Lab*, January 1992.

Millner, Amon. "Supporting Children As They Program to Make Physical and Virtual Objects Interact." In *Proceedings of the 7th International Conference on Interaction Design and Children*, 25–28. IDC '08. New York, NY, USA: ACM, 2008.

Minsky, Marvin. "Music, Mind, and Meaning." In *Music, Mind, and Brain*, edited by Manfred Clynes, 1–19. Springer US, 1982.

Monk, Andrew. "Mode Errors: A User-Centred Analysis and Some Preventative Measures Using Keying-Contingent Sound." *International Journal of Man-Machine Studies* 24, no. 4 (April 1986): 313–27.

Norman, Donald A. "Affordance, Conventions, and Design." *Interactions* 6, no. 3 (May 1999): 38–43.

Norman, Donald A. "Design Rules Based on Analyses of Human Error." *Commun. ACM* 26, no. 4 (April 1983): 254–58.

Novak, Joseph D. *Learning How to Learn*. Cambridge University Press, 1984.

Papert, Seymour. *The Children's Machine: Rethinking School in the Age of the Computer.* Basic Books, 1993.

Papert, Seymour A. *Mindstorms: Children, Computers, And Powerful Ideas.* New York: Basic Books, 1993.

Piaget, Jean. *The Origin of Intelligence in the Child.* Routledge & Paul, 1953.

Resnick, Mitchel. "Sowing the Seeds for a More Creative Society." *Learning and Leading with Technology,* December 2007, 18–22.

Resnick, Mitchel, John Maloney, Andrés Monroy-Hernández, Natalie Rusk, Evelyn Eastmond, Karen Brennan, Amon Millner, et al. "Scratch: Programming for All." *Commun. ACM* 52, no. 11 (November 2009): 60–67.

Resnick, Mitchel, Fred Martin, Robert Berg, Rick Borovoy, Vanessa Colella, Kwin Kramer, and Brian Silverman. "Digital Manipulatives: New Toys to Think with." In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems,* 281–87. CHI '98. New York, NY, USA: ACM Press/Addison-Wesley Publishing Co., 1998.

Resnick, Mitchel, and Eric Rosenbaum. "Designing for Tinkerability." *Design, Make, Play: Growing the Next Generation of STEM Innovators,* 2013, 163–81.

Resnick, Mitchel, and Brian Silverman. "Some Reflections on Designing Construction Kits for Kids." In *Proceedings of the 2005 Conference on Interaction Design and Children,* 117–22. IDC '05. New York, NY, USA: ACM, 2005.

Rusk, Natalie, Mitchel Resnick, Robbie Berg, and Margaret Pezalla-Granlund. "New Pathways into Robotics: Strategies for Broadening Participation." *Journal of Science Education and Technology* 17, no. 1 (February 1, 2008): 59–69.

Ruthmann, Alex, Jesse M. Heines, Gena R. Greher, Paul Laidler, and Charles Saulters II. "Teaching Computational Thinking Through Musical Live Coding in Scratch." In *Proceedings of the 41st ACM Technical Symposium on Computer Science Education,* 351–55. SIGCSE '10. New York, NY, USA: ACM, 2010.

Samuel Thibault, Chris Lyon. "MICK: A Constructionist Toolkit for Music Education," unpublished, 2001.

Sauer, Theresa. *Notations 21.* New York City, NY: Mark Batty Publisher, 2009.

Sawyer, R. Keith. *Improvised Dialogues: Emergence and Creativity in Conversation.* Westport, Conn: Praeger, 2002.

Schön, Donald A. "Designing as Reflective Conversation with the Materials of a Design Situation." *Research in Engineering Design* 3, no. 3 (September 1, 1992): 131–47.

Schön, Donald A. *The Reflective Practitioner: How Professionals Think in Action.* New York: Basic Books, 1983.

Seitz, Jay A. "Dalcroze, the Body, Movement and Musicality." *Psychology of Music* 33, no. 4 (October 1, 2005): 419–35.

Silver, Jay. "Awakening to Maker Methodology: The Metamorphosis of a Curious Caterpillar." In *Proceedings of the 8th International Conference on Interaction Design and Children*, 242–45. IDC '09. New York, NY, USA: ACM, 2009.

Silver, Jay, and Eric Rosenbaum. "Gifts for Intertwining with Modern Nature." In *Proceedings of the 9th International Conference on Interaction Design and Children*, 340–43. IDC '10. New York, NY, USA: ACM, 2010.

Stravinsky, Igor. *Poetics of Music in the Form of Six Lessons*. Revised edition. Cambridge, Mass.: Harvard University Press, 1959.

Suchman, Lucy A. *Plans and Situated Actions: The Problem of Human-Machine Communication*. Cambridge, Cambridgeshire ; New York: Cambridge University Press, 1987.

Tversky, Barbara. "What Do Sketches Say about Thinking." In *2002 AAAI Spring Symposium, Sketch Understanding Workshop, Stanford University, AAAI Technical Report SS-02-08*, 148–51, 2002.

Upitis, Rena. "The Development of Invented Music Notations: A Cousin to Invented Spellings." In *Arts and Learning Research*. The Journal of the Arts and Learning Special Interest Group of the American Educational Research Association, 1991.

Urbach, Martin. "A Manifesto for Ethical Music Education | Martin Urbach," September 25, 2014. http://www.martinurbach.com/2014/09/a-manifesto-for-ethical-music-education/.

Walter, W. Grey. "An Imitation of Life." *Scientific American* 182, no. 5 (May 1950): 42–45.

Wiener, Norbert. *Cybernetics, Second Edition: Or the Control and Communication in the Animal and the Machine*. Second edition edition. New York: The MIT Press, 1965.