Physical Programming: Designing Tools for Children to Create Physical Interactive Environments

Jaime Montemayor, Allison Druin
Allison Farber, Sante Simms, Wayne Churaman, Allison D'Amour

Human-Computer Interaction Lab, Institute for Advanced Computer Studies
University of Maryland, College Park, MD 20742 USA
monte@cs.umd.edu, allisond@umiacs.umd.edu

ABSTRACT
Physical interactive environments can come in many forms: museum installations, amusement parks, experimental theaters, and more. Programming these environments has historically been done by adults, and children have been the visiting participants offered a few pre-created choices to explore. The goal of our research has been to develop programming tools for physical interactive environments that are appropriate for use by young children (ages 4-6). We have explored numerous design approaches over the past two years. Recently we began focusing on a "physical programming" approach and developed a wizard-of-oz prototype for young children. This paper presents the motivation for this research, the evolution of our programming approach, and our recent explorations with children.

Keywords
Children, educational applications, programming by demonstration, ubiquitous computing, tangible computing, physical programming, physical interactive environments.

INTRODUCTION

Physical Interactive Environments
Researchers have found that a critical part of a child's early cognitive development is in negotiating the physical world [6, 7, 22]. Children can learn from building with blocks, drawing on paper, and even building make-believe worlds from boxes and bags. These constructive processes are how children can make sense and refine their mental models of the world [4, 22, 15]. Today with the use of emerging technologies, children can also manipulate images, sound, video or even robotic objects. These experiences can be saved and replayed, shared and constructed, even across geographically distant locations [24]. With emerging embedded and wireless technologies, these interactive experiences are no longer restricted to the use of keyboards, mice, and desktop boxes. Physical interactive environments can come in many forms: museum installations, amusement parks, experimental theaters, and even public toilets. As early as the 1960s, institutions such as the Exploratorium in San Francisco have been developing ways for visitors to learn about scientific and mathematical concepts through physically interactive experiences [27]. Many other museums now offer children the ability to explore such varied subjects such as music, at the Eloise W. Martin Center in Chicago, Illinois, and animals, at a working farm, at the Macomber Farm in Framingham, Massachusetts [26]. University researchers have also been developing physical interactive spaces. While this research has generally been developed for adult audiences, it has become more common to focus on children as users (e.g., NYU's Immersive Environments [10], MIT's KidsRoom [4], and the University of Maryland's StoryRooms [2]).

The enabling technologies that support these computationally enhanced physical environments can be found in the research of ubiquitous computing [31], augmented reality [18], tangible bits [16] and graspable user interfaces [12]. They share similar technical challenges, in scale, context awareness, gesture recognition, networking, location tracking, and software infrastructure [1, 25]. Equally challenging has been the introduction of these technologies into classrooms, due to the need for costly equipment, complicated authoring tools, and large space requirements [2].

There have in recent years been some examples of classroom technologies that enable children to learn from the construction of "computational objects to think with" [22, 23]. Seymour Papert and Mitchel Resnick, from the MIT Media lab have spearheaded research that has lead to such influential systems as the Logo mechanical turtle [23], a physical turtle that is programmed by a child to move around a room, and the LEGO Mindstorms Robotic Invention System [19], LEGO pieces that enable children to build robotic structures with sensors and actuators. More recently developed under the direction of Hiroshi Ishii at MIT, Curlybot [13] was created to mimic the actions of a child and encourage mathematical learning from physical play with a simple robotic sphere. And Tim McNerney's

Tangible Computation Bricks [20] enables young programmers to manipulate and connect physical action blocks that can react to sensor inputs. Storytelling systems have also explored the use of alternative physical interfaces for children. These include MIT's SAGE (Storyteller Agent Generation Environment) [30] which uses a stuffed rabbit, TellTale [3] which uses a plastic worm, the University of Maryland's PETS (Personal Electronic Teller of Stories) which uses various pieces of robotic stuffed animals [11], and Microsoft's Actimate Barney [29] with uses a stuffed doll. What each of these technologies has in common is that they offer a child the ability to shape their learning experience with an object that is computationally enhanced, yet physically familiar. While all are compelling technologies for children, they do not offer young people the ability to program an entire room or physical interactive space, children are merely in control of one object.

Programming Physical Environments

We have found that most physical interactive spaces are the result of adults' imaginations, not children's. Children are generally only able to choose between a few pre-created choices as participants in an experience. It is as if we adults only allow children to read books, but never allow them to tell their own stories. There is educational value in reading what others have written, but the act of authoring can offer children creative, problem-solving opportunities that are also critical to their cognitive development [15, 24]

Therefore, when we began our research two years ago in developing enabling technologies for physical interactive environments, our research priority was to support children as storytellers and builders from the very start of their physical experience. Out of this work came "StoryKits" a set of tools that would support the creation of what we now call "StoryRooms" [2]. When we first began our research in this area, we built an example StoryRoom experience based on the Dr. Seuss book, *The Sneetches* [14]. What this experience showed us was that we needed more generalized tools to develop our room-sized stories. We needed sensors and actuators that could easily augment any physical object, not just special "computerized" ones. We found that the children wanted to tell their interactive stories with props they created (e.g., out of cardboard boxes) (see Figures 1, 6, 8), and found objects they placed around a room (e.g., stuffed dolls, chairs, etc.). What we also found was that we needed a way to easily program this kind of environment, without taking the child away from their physical world and the act of storytelling. By asking children to work with programming technologies that were screen-based, their physical storytelling became secondary in importance to negotiating abstract programming languages.

Very little literature in this area has suggested an approach for novice users or children to physically create ubiquitous computing experiences. Novice user programming systems have historically focused on the traditional desktop computer model.

Figure 1: Creating props for a StoryRoom.

However, useful insights can be found in the research on visual programming languages (VPL), in particular, "programming by demonstration" systems (PBD) [21, 28]. Two strong examples for children are ToonTalk and KidSim (the commercial product now called StageCast Creator). KidSim enables the child programmer to define visual production rules, through comic strip like picture frames [8]. In ToonTalk [17], computational abstractions are replaced by concrete and familiar objects. A ToonTalk program is a *city* that contains *houses*. *Birds* fly between houses to transport messages. Houses contain *robots* that can be trained to accomplish small tasks. To program a robot, the programmer enters into its thought bubble to show it what to do. We have taken these examples of concrete demonstration, and considered how to physically define states and transitions for computational objects in a physical interactive environment.

Here is a simple physical example: Every time a child steps on a certain rug in her bedroom, she wants that desk lamp in the room to turn on. To demonstrate her intentions, she might perform the following sequence of physical activities: 1) invoke the programming mode, 2) step on the rug, 3) turn on the light, and 4) turn off the programming mode. In essence, by touching objects in the room, a child can create a programming statement, or rule.

This method of authoring or programming rules for physical interactive environments, we have come to call "Physical Programming," and can be defined as: *The creation of computer programs by physically manipulating computationally augmented (or aware) objects in a ubiquitous computing environment.*

In the remaining sections of this paper, we will present the evolution of our programming technologies and our recent explorations with young children (ages 4-6 years old). The implications of this research will be discussed as it relates to future physical programming directions.

THE EVOLUTION OF OUR PROGRAMMING TOOLS

Figure 2: Children on our team using art supplies to create low-tech prototypes of our programming tools.

Our ideas about the technical requirements as well as the user interactions of physical programming evolved over time. Our research team did not have as its initial goal to program without some visual display. We wanted to enable children to become authors of their own physical interactive environments, but we had few preconceived notions of what directions we would follow. However, thanks to our countless brainstorming sessions with children as our design partners (ages 7-11 years old) our notions of physical programming took shape. Over a two-year period we sketched ideas, created low-tech prototypes (see Figure 2), did walk-thru scenarios (see Figure 3), and developed mid-tech or wizard-of-oz prototypes (see Figures 4-8). Two afternoons a week, and two weeks over the summer our team of computer scientists, educators, artists, engineers, and children has met as design partners [2, 8] to work on this project and others.

Figure 3: Children on our team conducting walk-thru scenarios; the child on the right is wearing a "tell a story" crown on his head, placed there by the child on the left.

We went from building our own "StoryRoom" with specialized hardware (our Sneetches room), to developing numerous "StoryKits" that included various approaches to programming the physical environment. Our initial ideas suggested that children use a visual programming language on a screen in a box that sits on the floor (see Figure 4). This screen-based programming language looked similar to the large stuffed sensors and actuators that children physically placed on objects they wanted to become "magic" in the room (see Figure 5 for an example). Back in 2000, we believed:

"…*that a carefully designed visual programming system (should) enable children to author their own StoryRoom. Also, the programming system should provide a visualization of the story such that one can follow the storyline by looking at the visualization. Considering that the programming will be based on real objects in the physical world, the visual programming system should use notions that closely match those of the physical world*" [2].

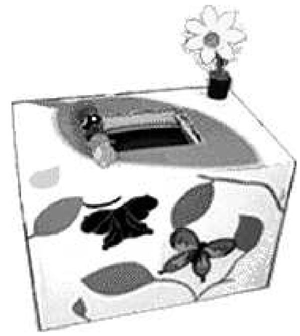

Figure 4: Prototype table for a screen-based programming approach for StoryRooms.

We no longer believe this to be the case. In our work with children (ages 4-9) we found that they had a hard time conceptually connecting what was on the screen with what they used in the physical room. Therefore, we finally just did away with the screen entirely and explored only the use of the physical sensors and actuators (which we have since come to call "physical icons" [16]).

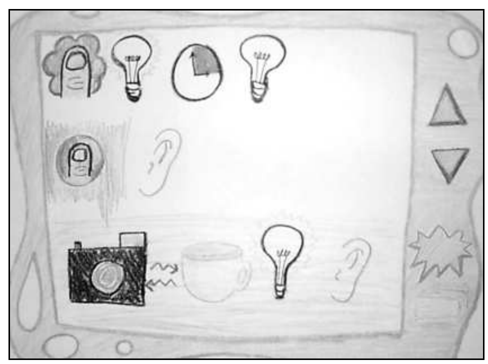

Figure 5: Some early visual programming ideas. The top line means "press the flower and the light stays on for 15 seconds." The third line means "when the camera and the cup are near each other, the light comes on and the ear will listen."

Figure 6: A large stuffed "hand" sensor that was placed on a child-created camera. When the hand was placed on the camera it could be physically programmed to be "touch sensitive". Therefore if a child touched the camera, a video might appear on the wall with a picture from the camera.

Ultimately the conceptual StoryKit that emerged not only included sensors and actuators represented as physical icons, but a "magic wand" which would signal the start and end of programming (see Figure 7). We found in our continual work with children that they needed some way to distinguish when they were programming and when they were using the actual physical icons as a participant in a story. Therefore for example, a child could place a "hand" near, or on a teddy bear. Then she could place a "sound box" next to a large pile of blocks in the corner. Then by tapping the magic wand on the hand then on the sound box, the room would be "programmed" to play a sound "Come here…" every time the hand was pressed.

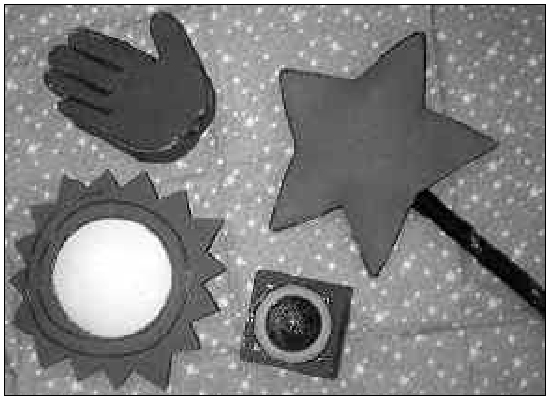

Figure 7: Physical Icons for Programming a StoryRoom; From left to right: A "hand" to make an object touch-sensitive; a "light" to make an object "light up"; a "sound box" to attach a sound to an object; a "magic wand" to signal the authoring mode.

The wizard-of-oz prototype

In order to understand if young children who had not helped design our programming tools could use our approach, we developed a mid-tech or wizard-of-oz prototype for some formative evaluation. We thought it was important to have the flexibility to experiment with different behaviors from the technology depending on the user interaction. But we found from many low-tech design sessions that often the "wizard" (person) could not track the many concurrent activities in the environment and react appropriately. Therefore, we developed a software application, written in RealBasic on the Macintosh, that allowed the wizard to define and group action-reaction rules on-the-fly as the children were using the technology. The wizard software broadcasted serial data packets via a 433 MHz RF Transceiver connected to the serial port on a Macintosh laptop. These signals were then received by RF transceivers embedded in the physical icons and interpreted by BASIC Stamp Microcontrollers. Based on the data content, the microcontroller then could turn on and off activators such as lights, sounds, and buzzers. Our implementation supported one-way communication, so children pressing the sensors, or tapping the icons with the wand did not actually activate anything. Through a one-way mirror adult researchers observed the actions of a child, and sent the appropriate response from the computer. For example if a child pressed the hand and expected a light to come on, it would.

OUR EXPLORATION WITH YOUNG CHILDREN

By developing a flexible proof-of-concept prototype, we were able to explore three basic questions: (1) Can young children (4-6 years old) comprehend what a story is about in a physically interactive environment such as a StoryRoom? (2), Can they use or participate in an already created story in a StoryRoom? (3) Can they use physical programming to create a StoryRoom? To answer these questions, we used qualitative observation and data collection methods that will be further described in the sections that follow.

Sessions Structure

We began our explorations with young children, by inviting four children in pairs of two (ages 5-6) to our lab to initially explore the tools. We did not structure their use of the tools; rather we wanted to see where they led us. One adult facilitated each session, with four adults taking notes, seven other children (our weekly design partners) taking notes and periodically asking questions, and one design partner child video taping the experience. From these sessions, one child design partner (age 11) wrote, "I don't think they got it when we started. When I showed them something it made sense then. I think it was good when they did it with me. Then they had some good ideas to show us."

With observations such as these, we quickly realized that we needed to structure the children's exploration at the start of their sessions with us. The notion of a physical interactive environment is conceptually difficult to understand and still somewhat uncommon, so to start off with the idea of programming one was difficult to grasp for children (and many adults). Therefore the four sessions that followed these initial sessions contained three parts: (1) *children as audience*, an adult tells an example story with a StoryRoom. (2) *children join adults as storytellers*, the children retell the story, so that they get to play with the props and squeeze the physical icons. (3) *children as physical programmers*, children are shown how to program with the physical icons and are asked to make up a new story.

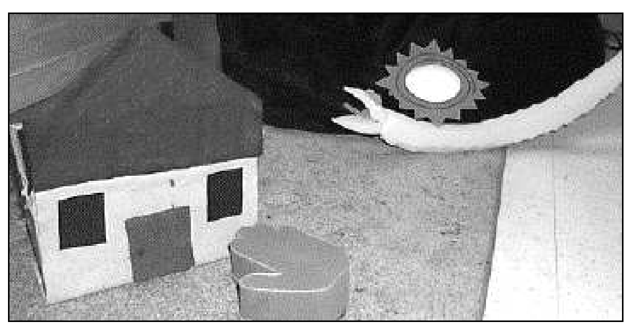

Figure 8: Examples of props and physical icons used in the "Irene story. The cottage and a hand icon are in the foreground. The snake and a light icon are in the back.

The sample story we used in our sessions was as follows:
One day, Irene was hiking in the woods behind her house, and she went farther than ever before. She became lost.

Irene saw a cottage just up ahead. She walked up to the cottage and saw a strange purple hand. She pressed the purple hand. (A purple light placed next to a furry mouse lights up.) She walks up to the purple light, and sees a mouse. She said, “Mr. Mouse, do you know a way back to my house?” Mr. Mouse replied, “I do not know where you house is. Maybe you should ask Mr. Koala.” Irene finds and goes up to Mr. Koala. She sees a green hand next to it. So she squeezes it and asks, “Mr. Koala, do you know the way to my home?” (A green light placed next to a snake lights up.) Mr. Koala said, “I do not know where your house is. Maybe you should ask Mr. Snake.” Irene follows the green light and sees Mr. Snake. She asks the same question. Finally, Mr. Snake says, “Sure, I know just the way. Come, follow me back to your home”

A default set of interaction rules which were used for the physical icons:

- The magic wand is only used for programming activities.
- The glow-fiber and buzzer of the icons indicate the *selected* state of the icon, and are used during the programming mode. For example, when the wand touches a light, its glow-fiber will blink. In addition, the icon will make a buzzing sound, its glow fiber blinks, and its light will turn on.
- To create a relationship between the hands, the lights, and the sound box, a child would take the magic wand, and tap the objects to create a group. For example, a group contains a purple hand, a green light, and the red side of the sound box. This means that whenever the purple hand is touched during the play mode, the green light will come on and the sound associated with the red side of the sound box will play.
- To start a story, put away the magic wand.

Session Participants

We conducted four subsequent sessions with the structures described above, at a local pre-school close to the University labs. In total, we were able to work with 11 kindergarteners (ages 4-6). Seven were boys, four were girls and each group included one girl and at least one boy. The first three groups had three children participating and the last group had two children. The first three groups worked with researchers an average of 13 minutes/session, and the last group worked for 50 minutes to see if we saw obvious differences in a longer time with the children.

Our research team was composed of five people: two adults who facilitated the storytelling with the children; one videographer in the room; one researcher situated behind a one-way observation window using the computer to react to what the children did; and one assistant, who helped interpret the children’s activities when they became difficult to see or understand.

Data Collection

We captured the activities and dialogue of all children with one video camera located in the classroom, about fifteen feet away from the story area. In our lab, we reviewed the tapes and created a contexual inquiry chart based on the *Cooperative Inquiry* methods described in [9]. We noted the time, verbal discussion, and activities in columns (see Table 1 for example).

Time	Quote	Activities
32:23	F: can you tell a story with these things?	W: yeah
32:44	W: I want to be mouse, B: I want to be koala	W grabs mouse, B grabs koala, G grabs snake.
32:57	W: the mouse went...	W grabs purple set and moves to the cottage
33:07		W positions the purple hand and light by the cottage. B holds on to the green hand.
33:13	W: the mouse went to sleep one night	W: touches the purple hand, the light came on
33:15		B: squeezes the green hand
33:23	W: who's on my door	
33:55		B: squeezes the green hand. Green light came on.

Table 1: Sample data from our contextual inquiry chart.

ANALYSIS OF OUR WORK WITH CHILDREN

After a review of the dialogue and activities, three members of the team together analyzed the data charts and developed “roles” (who a child was during a specific action (e.g., experimenter, story participant, etc.) and “activity patterns” (e.g., storytelling, playing, etc.). Once the team agreed on the initial codes for roles and activity patterns, then all the charts were coded. In Charts 1 & 2 (see below), the frequency of these roles and activity patterns were summarized for the last third of each session. It was decided by the team that during the third part of the session was really when the children were most in control and had the most freedom to explore. During the first two parts of the session, they were learning as much about the technologies as they were anything. In the sections that follow, we will discuss what we observed in the four sessions that were video taped and analyzed.

Children as audience

In this initial part of the session that lasted on average less than 2 minutes, children were shown the “Irene story” and we found that across the four sessions, children were quite attentive. They were fascinated by the use of the physical icons to create a physical interactive experience. At no time did any children look bored, instead many of the children could not wait to use the physical icons themselves to try out the story experience.

Children join adults as storytellers

During this section of the session, we found that most of the children (10 out of 11) were readily able to recall and reenact elements of the story. They actively participated in the StoryRoom experiences of Irene. Many of them (9 out of 11) also seemed to understand how to use the physical icons to participate in the story. Interestingly, one child began to experiment with the physical icons' behavior during this part of the session. She kept pressing on the hand to see if it would repeat turning on a light.

Children as Physical Programmers

During this third and final part of the session, the children were shown how to physically program and they explored the use of these technologies for storytelling. Our analysis of the roles and activity patterns revealed that the children spent most of their time experimenting with the tools (see Charts 1 & 2). They were not afraid to try out different combinations of taps with the magic wand, and frequently pressed the hand to explore the possibilities of what it affected. There were times when a technical glitch (e.g., the researcher at the computer sent the wrong command to the physical icons, or was delayed in responding), which also prompted the children to continue to experiment with the physical icons. Interestingly, we found that some of the children either waved the wand several times, or tapped repeatedly, until they saw the feedback they expected. Overall, in each session at least one child was able to form a definite idea about how to physically program with the tools.

Where the children seemed to have the most challenges with physical programming was in understanding the difference between the programming mode and the participation/use mode. The children understood that the wand helped them "make things magic" but they had difficulty understanding that it wasn't telling the story yet, merely getting it ready for others to hear it. We believe that this confusion may partially come from the feedback of light and sound when the children were in programming mode. As the children touched the physical icons with the wand, a sound would occur and a glow light on the icon would turn on. Many children were quite excited by this and thought this "was the story". We believe that perhaps by reducing the "excitement" of the feedback, that they may be more likely to see this as one step in the storytelling process.

In regards to storytelling, we found that the children told stories in three ways: (1) completely verbal with the use of no props or physical icons; (2) with the use of some props such as stuffed animals and verbal descriptions; (3) with the use of physical icons and props and verbal description. As Chart 2 summarizes, when the children were asked to tell a story, they most frequently just verbally told a story. The children fell back into what they knew best. However, once the researcher asked of they would like to use the things in the room to tell a story, they most frequently used both the physical icons and the props to physically program. Surprisingly, it was far less frequent for the children just to use the props.

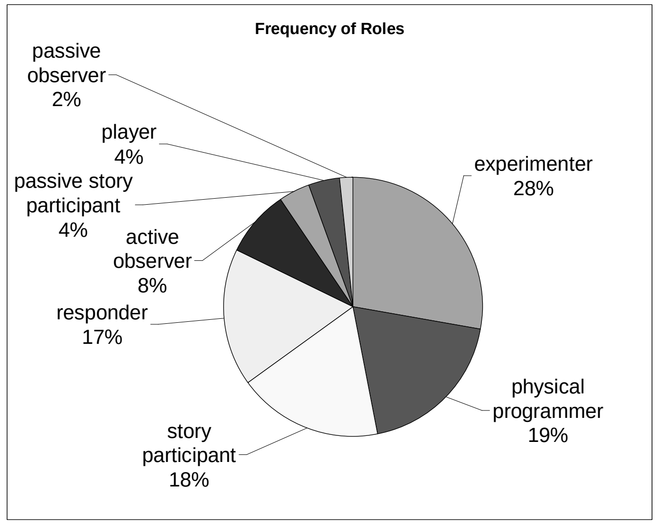

Chart 1: Frequency of children's roles during the last section of the session.

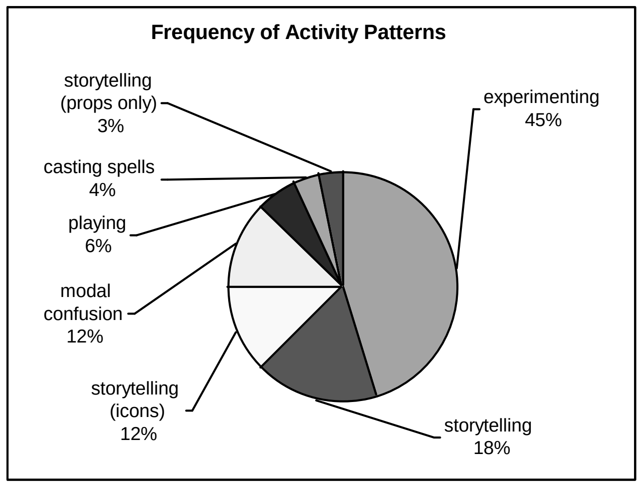

Chart 2: Frequency of children's activity patterns during the last section of the session.

The kinds of stories the children told were very similar to the Irene story they heard. In many cases only one or two elements were changed to make it their own. However, there were interesting additions to the stories they told. For example, one child incorporated the physical icon lights as decorations on a cottage prop. In her story she had the characters ask, "Who is there? Would you please turn off the lights? I need to sleep." We believe that perhaps, had there been additional props (outside of the ones used for the Irene story) and more time to explore, more original stories might have emerged.

LESSONS LEARNED

In understanding what we have learned with children, we refer back to our three initial questions: (1) Can young children comprehend what a story is about in a physically interactive environment such as a StoryRoom? (2), Can they use or participate in an already created story in a StoryRoom? (3) Can they use physical programming to create a StoryRoom?

With regards to the first question, we saw without a doubt that children ages 4-6, who had no experience in designing our technologies, can easily comprehend what the story is about. The interactivity in the StoryRoom did not get in the way of understanding the Irene story. We also saw with regards to the second question, that all of the children could also use or participate in an already created story. Once shown how to interact with the physical icons, they had no trouble interacting with the StoryRoom experience.

As for the third question concerning physical programming, the answers are less clear cut. We did see in each session one or more children able to physically program. They understood that placing the physical icons on a prop around the room either offered some input or output. They also understood that the physical icons had relationships to each other based on how they were programmed. In fact, out of the 11 children we worked with only 3 children could not comprehend any aspect of this approach. Thanks to a longer session with the last group, we now believe that had we spent a longer time with each group, all of the children would have been able to accomplish physical programming. But considering the short period of time we were with the children, they were able to accomplish much more than we expected in some ways. It is not surprising that their main difficulty was in understanding the difference between programming and participation in an already created story. At this young age, children's most common form of storytelling is improvisational storytelling (many times referred to as "play") where children freely move in and out of storytelling and "storylistening" [2]. We now believe this may be our biggest challenge in supporting children with physical programming. Is there a way to naturally move between programming and participating? The magic wand may be only part of the solution.

In regards to lessons learned about our methods, we believe that the mid tech or wizard-of-oz prototype served us well. It went a long way in simulating the full experience of physical programming. It offered us a flexible way in exploring our ideas with children, without having to spend many more months fully developing the technologies. We also believe that without the numerous low-tech prototyping sessions, scenario walk-thrus, and initial observations with children, we could not have been as successful as we ultimately were.

FUTURE WORK

Our work is now focused in two areas. The first is incorporating location-aware technologies, as well as implementing better communication protocols, in order to minimize human intervention (wizard). For our design process, having a human in the loop was critical for on-the-fly changes, but we now better understand the behaviors necessary for the technology to perform. The addition of location-aware (e.g. infrared, ultrasound, WaveID RF tag) technology into the physical icons would enable us to track physical programming events. We are currently working on faster two-way communication, so that latency does not confuse the children. Further down the road, we will address some other scaling issues: what devices, and how many, will children need to be expressive. We will also consider how children might dictate additional programming intentions such as looping and timing.

Our other area of focus is in the design process, and evolving our methods with children. We are currently planning more brainstorming sessions with our child design partners in the lab. We will also be continuing our collaboration with the young children of a local daycare facility. We are working with their teachers to develop a classroom integration plan for our physical programming technologies. In this way we hope to understand what young children can do with these technologies over months of time. We want to understand what programming approaches they take, and what storytelling experiences they develop.

ACKNOWLEDGMENTS

This work has been funded by the European Union's i3 Experimental School Environments initiative, DARPA, and the Institute for Advanced Computer Studies. We would also like to thank our colleagues in the Human-Computer Interaction Lab and the incredible children on our team, who continue to inspire, challenge, and support our research efforts.

REFERENCES

1. Abowd, G. D., and Mynatt, E. D. Charting past, present and future research in ubiquitous computing. *ACM Transactions on Computer-Human Interaction, Special issue on HCI in the new Millenium 7*, 1 (March 2000), 29–58.

2. Alborzi, H., Druin, A., Montemayor, J., Platner, M., Porteous, J., Sherman, L., Boltman, A., Tax´En, G., Best, J., Hammer, J., Kruskal, A., Lal, A., Plaisant-Schwenn, T., Sumida, L., Wagner, R., and Hendler, J. Designing StoryRooms: Interactive storytelling spaces for children. In *Proceedings of Designing Interactive Systems (DIS-2000)* (2000), ACM Press, 95–104.

3. Annany, M., and Cassell, J. Telltale: A toy to encourage written literacy skills through oral storytelling. Presentation at *Conference on Text, Discourse and Cognition* (Winter 2001).

4. Assey, J. The Future of Technology in K-12 Arts Education. White Paper for the U.S. Department of Education, Requested as a result of: *Forum on Technology in Education: Envisioning the Future.* (2000), http://www.air.org/forum/Assey.pdf.

5. Bobick, A., Intille, S. S., Davis, J. W., Baird, F., Pinhanez, C. S., Campbell, L. W., Ivanov, Y. A., Schutte, A., and Wilson, A. The kidsroom: A perceptually-based interactive and immersive story environment. In *PRESENCE: Teleoperators and Virtual Environments* (August 1999), 367– 391.

6. Brosterman, N. *Inventing Kindergarten*. Harry N. Adams Inc., 1997.
7. Bruner, J. *Toward a theory of instruction*. Harvard University Press, 1966.
8. Cypher, A., and Smith, D. Kidsim: End-user programming of simulations. In *Proceedings of CHI 95* (1995), ACM Press, 27–34.
9. xxxxxx, Cooperative inquiry: Developing new technologies for children with children. In *Proceedings of CHI 99* (1999), ACM Press, 592-599.
10. Druin, A., and Perlin, K. Immersive environments: A physical approach to the computer interface. In *Proceedings of CHI 94* (1994), ACM Press, 325–326.
11. Druin, A., Montemayor, J., Hendler, J., McAlister, B., Boltman, A., Fiterman, E., Plaisant, A., Kruskal, A., Olsen, H., Revett, I., Plaisant-Schwenn, T., Sumida, L., and Wagner, R. Designing PETS: A personal electronic teller of stories. In *Proceedings of CHI 99*(1999), ACM Press, 326-329.
12. Fitzmaurice, G. W., Ishii, H., and Buxton, W. Bricks: Laying the foundations for graspable user interfaces. In *Proceedings of CHI 95* (1995), 442–449.
13. Frei, P., Su, V., Mikhak, B., and Ishii, H. curlybot: Designing a new class of computational toys. In *Proceedings of CHI 2000, CHI Letters, 2*(1), (2000), ACM Press, 129–136.
14. Geisel, T. *The Sneetches, and other stories*. Random House, New York, 1961.
15. Given, N. & Barlex, D. The Role of Published Materials in Curriculum Development and Implementation for Secondary School Design and Technology in England and Wales. International *Journal of Technology and Design Education*,*11*(2), 2001, http://www.wkap.nl/sample.pdf?313033.
16. Ishii, H., and Ullmer, B. Tangible bits: Towards seamless interfaces between people, bits and atoms. In *Proceedings of CHI 97* (1997), ACM Press, 234–241.
17. Kahn, K. Generalizing by removing detail: How any program can be created by working with examples, 2000. Available at http://www.animatedprograms.com/PBD/index.html.
18. Mackay, W., Velay, G., Carter, K., Ma, C., and Pagani, D. Augmenting reality: Adding computational dimensions to paper. *Computer-Augmented Environments: Back to the Real World. Special issue of Communications of the ACM 36*, 7 (1993).
19. Martin, F., Mikhak, B., Resnick, M., Silverman, B., and Berg, R. To mindstorms and beyond: Evolution of a construction kit for magical machines. In *Robots for kids: New technologies for learning*, A. Druin and J. Hendler, Eds. Morgan Kaufmann, San Francisco CA, 2000, 9-33.
20. McNerney, T. S. Tangible programming bricks: An approach to making programming accessible to everyone. *Master's thesis*, MIT Media Lab, 2000.
21. Myers, B., and Buxton, W. Creating highly-interactive and graphical user interfaces by demonstration, computer graphics 20(3). In *Proceedings of SIGGRAPH '86* (1986), 249–258.
22. Papert, S. *Mindstorms: Children, computers and powerful ideas*. Basic Books, New York, 1980.
23. Resnick, M., Martin, F., Berg, R., Borovoy, R., Colella, V., Kramer, K., and Silverman, B. Digital manipulatives: New toys to think with. In *Proceedings of CHI 98* (1998), ACM Press, 281–287
24. Roschelle, J. M, Pea, R. D., Hoadley, C. M., Gordin, D. N., & Means, B. Changing How and What Children Learn in School with Computer-Based Technologies. *The Future of Children: Children and Computer Technology*, *10*(2), (Fall/Winter 2000), http://www.futureofchildren.org/cct/cct_04.pdf.
25. Salber, D., Dey, A., and Abowd, G. Ubiquitous computing: Defining an hci research agenda for an emerging interaction paradigm. Tech. rep., Georgia Institute of Technology, (1998), *Tech. Report GIT-GVU-98-01*.
26. Edwin Schlossberg Incorporated. url: http://www.esidesign.com/.
27. Semper, R. J. Science museums as environments for learning. *Physics Today* (November 1990), 50–56.
28. Smith, D. C. Pygmalion: An executable electronic blackboard. In *Watch What I Do: Programming by Demonstration*, A. Cypher, D. C. Halbert, D. Kurlander, H. Lieberman, D. Maulsby, B. A. Myers, and A. Turransky, Eds. MIT Press, 1993, ch. 1.
29. Strommen, E. When the interface is a talking dinosaur: Learning across media with actimates barney. In *Proceedings of CHI 98* (1998), ACM Press, 288–295.
30. Umaschi, M. Soft toys with computer hearts: Building personal storytelling environments. In *Proceedings of CHI 97* (1997), ACM Press, 20–21.
31. Weiser, M. The computer for the twenty-first century. *Scientific American* (September 1991), 94-104.

.