Computers and Squeak as Environments for Learning

by John Steinmetz

Introduction

Computers, although promoted as an all-purpose educational panacea, certainly aren't. The machines do have potential to assist certain learners in specific ways. Computer scientists, software engineers, teachers, parents, and children are using Squeak to explore that potential.[1] Some of Squeak's features are designed (and are being designed) to help non-experts learn programming and to facilitate construction of environments and projects for learning about other subjects.

To promote more thoughtful discussion about computers and learning, and to provide some background before considering Squeak projects, this chapter will begin with general thoughts about children and computers. Part 1 presents some assumptions and persistent misconceptions about computers and learning.

Parts 2 and 3 will discuss two ways that computers can assist learning. Part 2 presents their most common current use, simulating older media—such as words on paper or musical sound—while offering extra leverage for working in those media. Part 3 considers brand new possibilities offered by computers, with entirely new ways to perceive and understand. Squeakers are developing tools, ideas, and genres to help these new media evolve.

The final "Parting Shots" are thoughts about project design and collaboration.

The illustrations throughout this chapter come from Squeak projects.

Part 1: Children, computers, and learning

Because computers seem so powerful and versatile, people sometimes

---

[1] The Disney Squeak team developed the infrastructure and ideas for Squeak-based learning environments. Alan Kay originated many of the general concepts and project ideas, and these have been realized and implemented brilliantly by Scott Wallace, John Maloney, Ted Kaehler, and Dan Ingalls. Kim Rose has organized experiments in classrooms and sought the system improvements needed to make them feasible. BJ Conn and Kathleen Brewer have designed projects specifically for children, and their work has helped the programmers to improve the system. Meanwhile, Mark Guzdial at Georgia Tech has been using Squeak with adult learners. Work with Squeak and learners is not confined to one continent; see the chapters on Mathmorphs in this volume.

talk as though using computers can guarantee positive outcomes. This misconception is especially prevalent in discussions of education.[2] In discussing any medium, it is important to remember that outcomes depend on the intentions and assumptions of the people using the medium. Skillful educators and learners can use almost any medium to advantage, but no medium can, by itself, determine what a person will learn.

Nevertheless, schools and households are buying lots of computers for children, often without a clear idea of what the machines are for, but convinced that they will help somehow. The result may be an increase in wasted time, boredom, and frustration for teachers, students, and parents.[3] Worse yet, time spent with computers by both teachers and students may deprive children of the attention and experiences they need in order to learn and grow.[4] Clearer thinking is needed about what computers are good for in education.

I should say right away that, despite the pleasure I have had in using computers for almost 20 years, and after nearly as many years of working with computers and schoolchildren in various research settings, I mostly keep my own children (age 4 and 11) away from computers.

I want my children to occupy themselves mainly with real stuff, and with virtual stuff from their own imaginations. They need many years of experience contacting the world through their senses and imaginations. I think they need to spend lots of time with older technologies and older kinds of virtuality like language, musical instruments, art supplies, magnifying glasses, books, basketballs, arithmetic, and so on. Children need to develop an inner life, the ability to visualize, imagine, and follow a chain of thought, because reading, mathematics, and other later work require not just the decoding of symbols but also the generation of inner spaces where the meaning of the symbols can take shape. Media of all kinds can disrupt a child's ability to form her own visual images, create her own stories, and form that crucial inner world.

Parents and teachers, concerned for children's futures, sometimes push children too soon into symbolic and abstract work. Politicians hope to improve reading scores by teaching reading earlier, but earlier is not necessarily better, and early childhood specialists maintain that the mentality of early childhood is not yet suited to sequential and abstract thought. The main work of early childhood is imaginative play. Nevertheless, many kindergartens now require homework, and I recently heard about a nursery school that devotes a certain number of hours per week to computer activities.

---

[2] See Todd Oppenheim's excellent article, "The Computer Delusion," in the July 1997 issue of the *Atlantic Monthly*, or at www.theatlantic.com/issues/97jul/computer.htm. The web version includes links to other information and organizations.

[3] In researching her book *Failure to Connect*, Jane Healy spent months visiting schools that were proud of their use of computers. She reported in a 1999 talk for Whole.org in Los Angeles that hardly any of these programs were contributing significantly to student learning.

[4] Lowell Monke, "Computers in Schools," *Yes!*, #8, Winter 1998-99, pg. 33.

Content aside, some investigators[5] observe that watching video screens may interfere with children's brain development, both by presenting harmful stimuli and by depriving the child of necessary movement and multisensory stimulation. Using computers (as well as television and video games) may harm young children even if the content seems to be beneficial.

After children have a lot of contact with the world, lots of imaginative play, and lots of practice using representation systems, they should be able to profit from the computer's abstractions and simulations. Computer work probably doesn't need to begin until 5th or 6th grade, or maybe later, when children's brains can withstand the neurological impact of screen stimuli and when their minds are ready to grasp cause and effect.

Students in a fourth- and fifth-grade classroom working with Squeak. The desks, with the computers inside and windows on top, can be used for computerless work as well. Open Charter School, Los Angeles. (Photo by Kim Rose.)

[5] Healy, Jane, *Endangered Minds: Why Children Don't Think and What We Can do About It*, 1990, New York: Simon and Schuster.

Buzzell, Keith, *The Children of Cyclops: The Influence of Television Viewing on the Developing Human Brain*, 1998, California: Association of Waldorf Schools of North America.

Pearce, Joseph Chilton, *Evolution's End: Claiming the Potential of Our Intelligence*, 1992, California: Harper San Francisco.

Winn, Marie, *The Plug-in Drug*, 1985, New York: Penguin Books.

[6] Dolores Patton, Leslie Barclay, Doreen Nelson, and I learned to ask this question in a research project of Apple Computer's Vivarium Program, a precursor to the Disney Squeak team.

But how should learners use computers? How can computers help? Educational use of computers is frequently colored by unexamined assumptions. Before considering how computers can help learners, let's take a look at how people *think* computers can help.

Examining assumptions about computers and learning

Computers will make learning fun.

Much of the excitement about computers in education has been generated by computer companies, but some of the enthusiasm comes from parents who have seen their children having fun with computers and computer games. I'm sure that many parents, remembering their own schooling as unhappy and boring, hope that computers will somehow enliven their children's education.

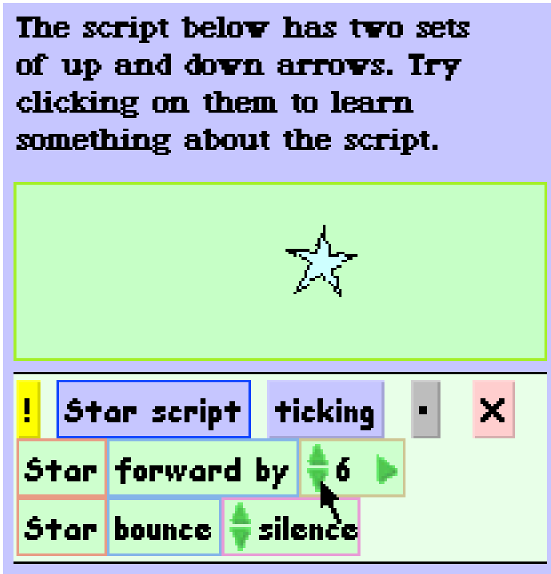
A child's ability to remain attentive to a computer game without becoming bored comes in large part from the constantly shifting visual stimuli which continually re-engage the child's attention, no matter what the content. This is the fun of being entertained—even entrained.

The fun of learning is of a different kind, having to do with making interior connections between something known and something new—the "aha!" experience. Entertaining stimuli may attract somebody to learn, but the real fun of learning is not the fun of being entertained. Entertainment can get in the way of learning if it prevents the learner from doing her part—engaging in a uniquely personal way with the content, with the activity, or with the learning process itself.

Learning requires knowing how to *devote* attention to something; repeatedly allowing one's attention to be *captured* does not teach how to direct and focus attention purposefully. As many teachers have noticed, students who are used to being entertained by video stimulation may be less able to engage, less able to pay the sustained attention required in order to enjoy the learning process.

Fun can be an important component in a learning environment, and fun is often a clue that learning is happening. One helpful question to ask about a learning activity is "Who's having the fun?"[6] In other words: who is making creative choices, who is thinking, who is having insights? In some learning situations, teachers think up interesting projects, find relevant information, discover insights, and share their discoveries, while students are merely expected to absorb the results. Those teachers are having most of the fun and doing most of the learning. It can be helpful to think about where the fun is in a subject or a learning activity, and how to share some of that fun with the students.

If computers were to make learning more fun, they would do so not with flashy effects, but by helping learners make personal contact with the heart of a subject, or by helping learners activate and develop their own abilities to perceive and understand.



A page from a computer activity introducing ideas through simple activities.

Computers will make learning more efficient.

In many fields computers have been used to reduce the human hours necessary to accomplish tasks. Computers have also taken on many repetitive jobs that formerly had to be done by people. It may seem logical that education could be similarly automated, that teachers and students using computers could teach and learn in less time, that computers could reduce the repetition needed for learning.

Unfortunately for this assumption, the main tasks in learning occur inside the learner, not in any place that can be automated. So far no computer system has been able to monitor the myriad factors—such as motivation, temperament, learning style, previous knowledge and skills—well enough to help a student learn more efficiently.

Computers do have a role in learning, but their role is not to reduce work time. In fact, computers may make learning more labor-intensive. The most effective computer-using classrooms I have seen require more teacher-hours than other forms of teaching, because students need so much help using the computers and computers need so much support time.

It is certainly possible to make learning more efficient and effective, not through automation but through better understanding of learning processes, and through heightened awareness of the motivations, skills, knowledge, and temperament of individual learners.

Computers will make learning easier

Behind the assumptions about fun and efficiency is an understandable hope that computers will make learning easier. This hope predates computers, of course, and much work has been done to help learners overcome unnecessary obstacles, whether interior obstacles like fear or incompatible learning style, or external obstacles like poor sequence or negative classroom atmosphere. Much of what has been learned in the last 50 or 100 years about such obstacles has yet to be integrated into mainstream education.

Nevertheless, no amount of removing obstacles can alter the basic truth that some things are difficult to learn. The difficulties need not involve suffering, but learning often requires time, patience, care, and persistence. Some especially worthwhile things are especially difficult: learning to play a musical instrument, learning to make a 3-point shot, learning calculus, learning to get along with people, learning to draw, learning to read, learning a foreign language.

Learning difficult things has a side-benefit, too: going through the process can help a person develop a capacity to work through difficulties.

In many cases, removing the hard parts removes the best parts. If computers make things easy by leaving out or glossing over the best parts, they help no one. If they make things easier by helping learners to focus on what's important and stick with it, they could be helpful. If they also remove some unnecessary obstacles, they could be a bigger help.

The main job of schools and teachers and others who help learners is to help learners through the worthwhile hard parts.

Computers will provide information.

Too often people use the word "education" as if it meant "providing information" and "learning" as though it meant "absorbing information." Of course that definition is woefully incomplete. Learners usually have access to plenty of information; they need to learn how and why to make use of it. Using computers to increase the amount of available information is not necessarily a help.

It should be obvious that, particularly for children, collecting information is less important than developing ways to transform information into knowledge and understanding. If computers can help with that, then their information-providing could be a positive contribution.

Computers will connect children with other people.

Computers can mislead adults as well as children about their ability to connect people meaningfully. Computer teacher Lowell Monke writes in *Yes!* magazine about a classroom of teenagers exchanging email with students in other countries. This special project, which was supposed to help the students develop understanding for people from different cultures, seemed not to help at all in the hallway outside the classroom, where other students from ESL classes, all recent immigrants, were ignored all year long by the students who were supposedly learning to transcend cultural barriers via email.[7] Let's not kid ourselves about what can be learned through correspondence—sometimes you have to tough it out through the difficulties of in-person contact.

As with other computer capabilities, their connectivity is neither positive nor negative. Everything depends on intent, the use to which the capability is put, and the extent to which the activity helps learners integrate what they learn.

The future will be technological, so our children need to know technology.

Until very recently, every computer innovator and everyone using computers at work had learned to use computers in adulthood. During these people's childhoods no computers were available to children, yet as adults they were able to invent, improve, and use computers. You don't need to have a computer as a kid to become a computer expert later in life.

Besides, becoming expert at today's technology will not be much help tomorrow. The most valuable job skills in the future will probably be in the areas of creative and flexible thinking, assimilation of new kinds of information, acquisition of new skills, and ability to recognize what's essential and what's fluff. As always, the advantage will go to those who have cultivated the human skills and knowledge for dealing with the inner and outer world. Learning to understand the purposes and uses of knowledge, skills, and technologies of all kinds—starting with language and crayons—will help children to create their future.

In every culture, people must learn how to use essential technologies. If computers were to help people prepare to use future technologies, they would do so by helping people to become less dependent on any particular technology, more adaptable to different approaches, and more savvy about advantages and pitfalls of any given technology.

Part 2: Computer simulations of older media

Although computers are probably inappropriate for young children, they can help older children, adolescents, and adults with learning. Computers can't solve all learning problems, but they can help in particular ways. For people who are developmentally and educationally prepared, work with simulations, virtual environments, and computational representations can play a role in their further growth.

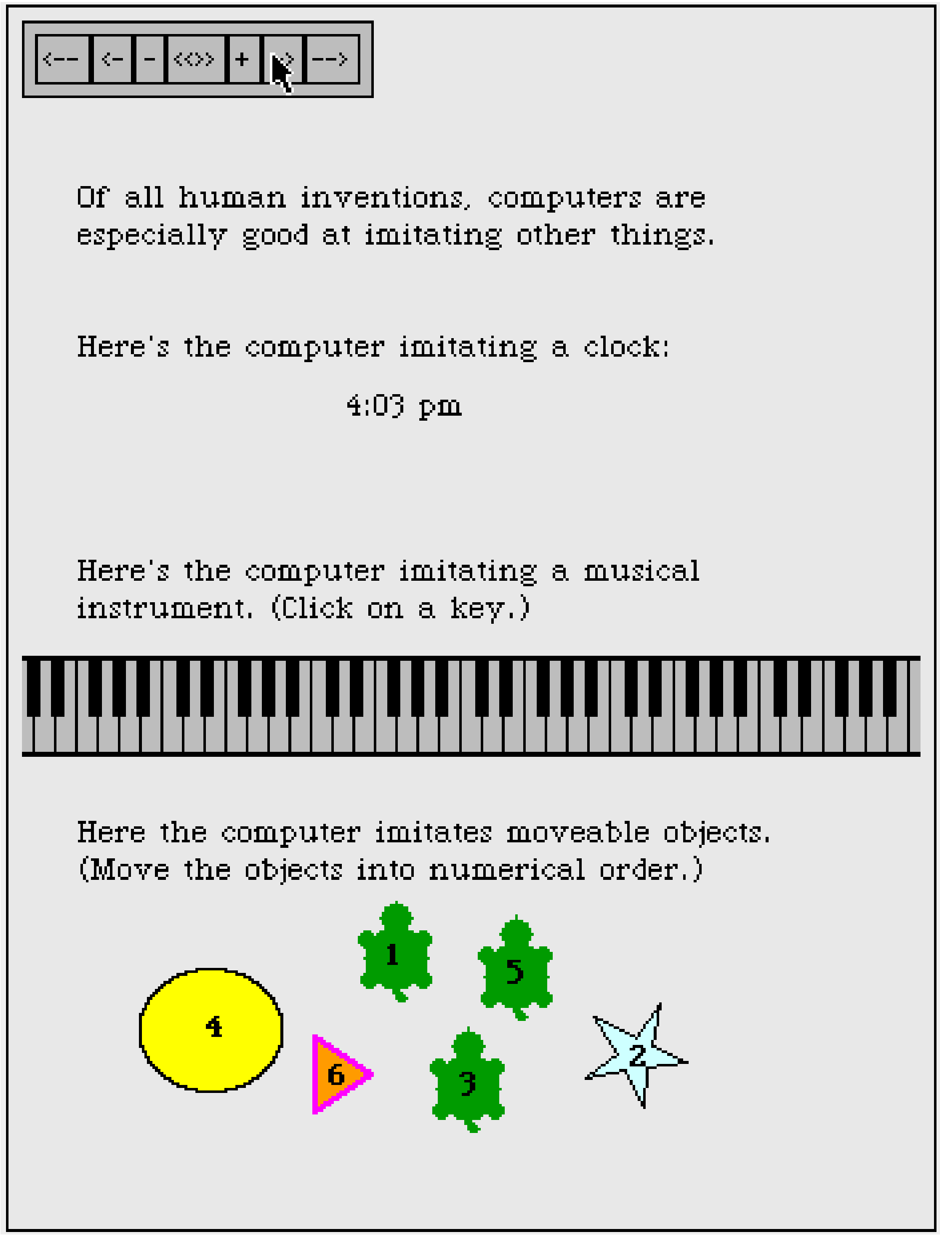
Computers can help learners in two general ways: by simulating older media or by presenting new media that are unique to computers. Parts 2 and 3 discuss these two approaches to learning with computers.

Computers are commonly used in education (and in workplaces and homes as well) to simulate other media that were in existence long before computers. Text editors simulate words on paper; drawing and painting

[7] Lowell Monke, "Computers in Schools," *Yes!*, #8, Winter 1998-99, pg. 33. This insightful essay by an experienced computer teacher is also posted at www.futurenet.org/8Education/monke.html.

programs simulate older forms of artmaking. A computer can simulate an ensemble of musical instruments, a model of the solar system, or an ant colony.

In the illustration below, a computer imitates a page in a book about how computers can imitate other media.

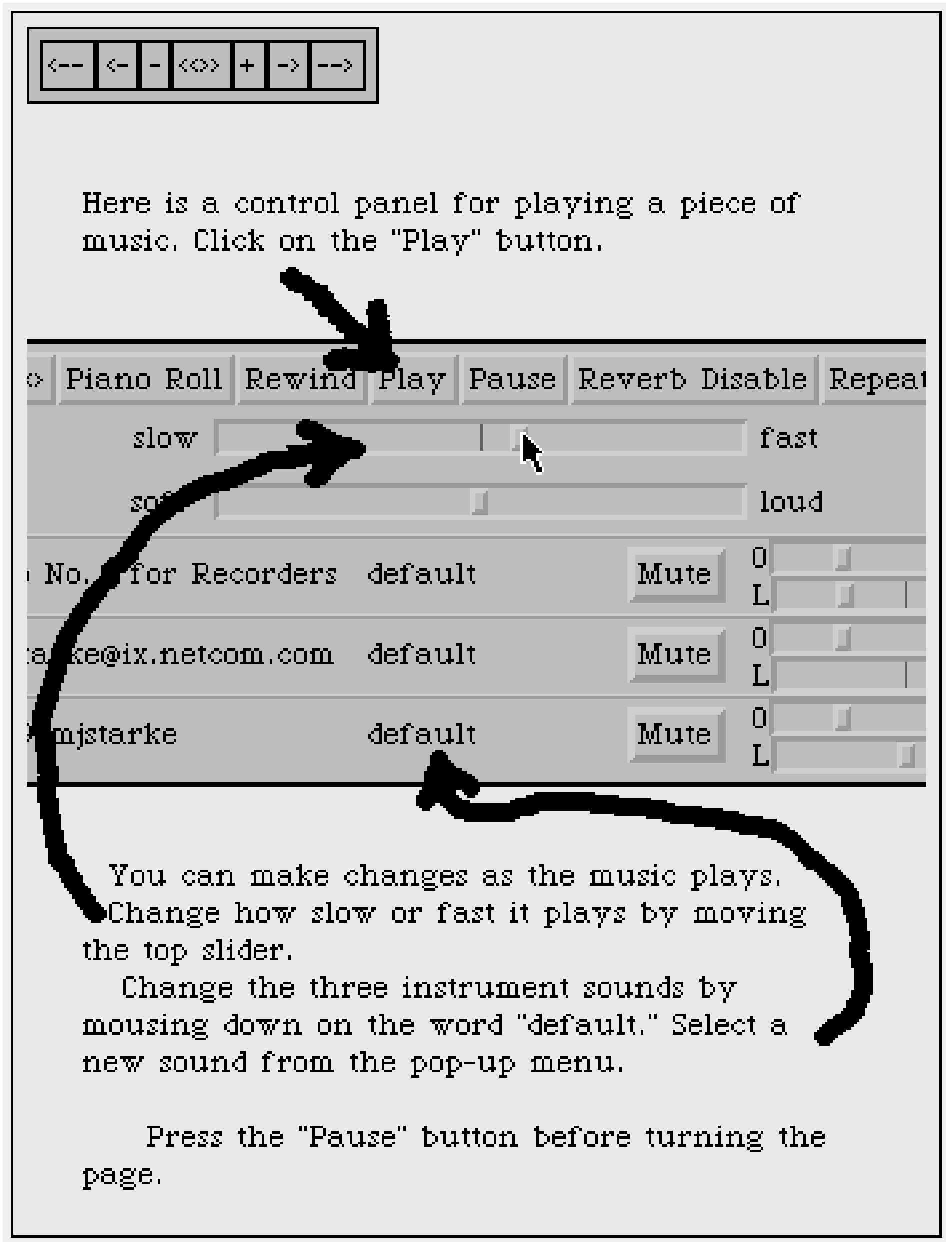


Potential advantages of simulating older media

Simulations of older media offer leverage.

Computers can open up possibilities that were not available in the original medium. Text editors allow rewriting without so much retyping or erasing.

Drawing programs allow a drawing to be altered and manipulated in ways that would be impossible on paper. Computer simulations of music give novices opportunities to try different orchestrations, change tempos, and otherwise alter the music without having to spend years practicing a musical instrument.



Novices can jump to an intermediate level.

If you want to compose music nowadays, you don't necessarily have to be a good performer. You can have a computer play your music for you. The computer may also offer you compositional assistance, so you don't have to know very much about composing either.

Computer simulations can facilitate learning by allowing novices to explore a medium at an intermediate level, without having to do the skill-building that used to be unavoidable. This means that a novice can start working immediately with essential elements of that medium, the elements that make that medium exciting.

Similarly, desktop publishing software makes all of us potential graphic designers, without the need for specialized equipment and training. Word processing software, by reducing the pain of erasing and retyping, can foster a healthy playfulness toward language and writing, making it easier to try out different ways to say something.

Learners may more easily contact "the good stuff."

Using computer simulations of older media can help novices contact the romance and excitement of a medium before investing in skill-building. Having an early experience of a medium's potential may inspire people to stick with the skill-building work.

Learners may be able to see the forest instead of the trees.

Some education methods pay a great deal of attention to skill building without fully exploring what the skills in that medium are for. This is like not being able to see the forest for the trees. For example, some music students become technically competent but unexpressive; some writing students learn to write correctly but not beautifully; some science students know scientific principles but don't know how to think like a scientist. One great promise of computers is that, by putting people in contact right away with some of the power and potential of a given medium, they may help learners stay aware of the forest while working to master the trees.

Computer simulations facilitate hands-on work with symbols.

Currently only a small percentage of adults understands science and mathematics well enough to participate in discussion of important issues facing our society. One reason so many adults have trouble with math and science is that they have trouble with the symbolic representation systems.

Computers might be able to help more people practice scientific and mathematical thinking by offering opportunities to work in different modes. Approaching the same concept or data visually and kinesthetically as well as symbolically might help more people become fluent with the symbols.

Pitfalls of simulating of older media

None of those potential advantages comes automatically. Unless the users are working toward that advantage, it won't come. And for every advantage that a technology offers, there is a flip side with a risk or a loss of some kind.

Here are some of the potential disadvantages of using computers to simulate older media:

Computers don’t guarantee seeing the forest.

Computers can’t automatically bring people to awareness of a medium’s higher-level pleasures. Teachers who are good at ignoring the forest may continue to do so even while using computers.

In the mid ‘80s, walking down an elementary-school hallway, I passed a classroom full of Atari 800s. I had been having lots of fun with my first computer, an Atari 800, learning the word processor and exploring LOGO. Knowing how good kids are at having fun, I was eager to see what they were doing in the computer room.

To my dismay, I found the room full of bored children assigned to an uninspiring task. On the walls were nicely lettered computer terms, and beside each monitor was a card with step-by-step instructions for typing a sentence that said, “My name is _____.” I didn’t detect any fun, nor any points of contact with the medium’s potential.

I shouldn’t have been surprised. For generations bad teaching has managed to transform thrilling matters —such as literature, history, and science—into boredom. This particular school had applied a similar transformation to its computers.

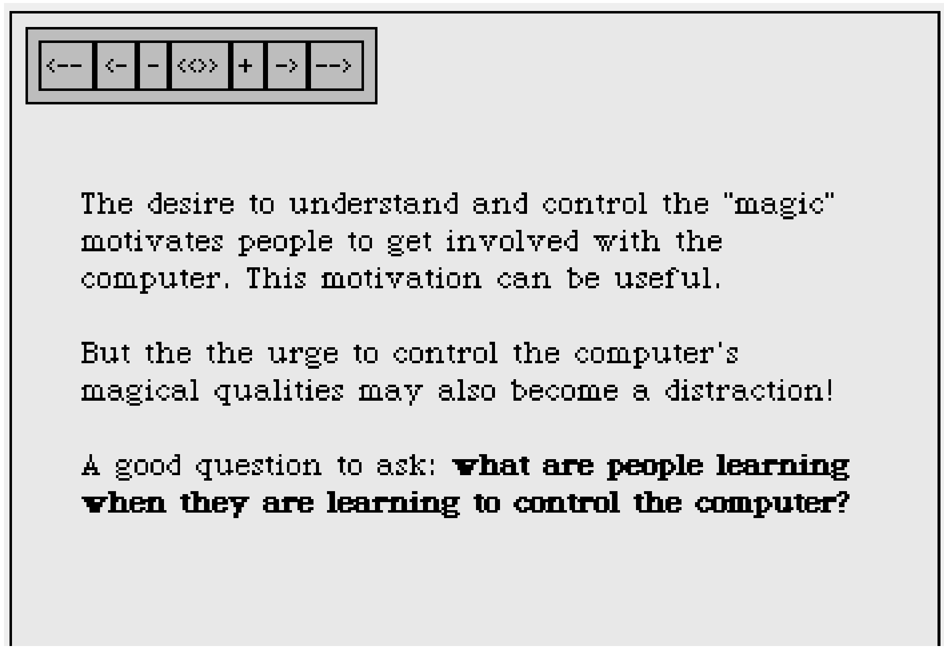
Computers present more trees.

When using computers to simulate an older medium, users may become distracted by features. For instance, using a computer to learn about writing requires that the focus be on the writing, not on the computer. The computer may help the learner to manipulate the writing, but it is sometimes hard to tell whether the learner is learning about writing or about the software.

This irony is not unique to computers: the very features that grant easier access to the forest also present more trees.

Power doesn’t necessarily lead to insight.

For novices, exercising intermediate-level powers may become distracting. Learning ultimately requires internal change, but making changes to computer representations is so much fun that it can keep going without much learning taking place.



The computer "page" above asks an important question for those who would use computers for learning. Are users learning useful skills that can be transferred to other domains? Are they learning skills that will grant them access to higher-level learning? If they're fiddling around, is it the kind of fiddling that eventually leads to insight? Or is it just fooling around?

Skipping skills may be limiting.

Without having gone through a skill-building phase, people may be less able to grasp and work with a field's fundamental issues. For instance, part of musicality is in the body, not in the mind, and that physical aspect of musicianship develops through learning to play an instrument or to sing. It might be hard to compose well without getting physical first.

Not everything can be discovered by messing around.

Good writing doesn't come about just through playing with words; a writer also needs to be aware of larger issues like pacing and structure, issues which are not explicit in words themselves, nor in word-processing tools. Similarly, one can play around with a page-layout program forever without stumbling across principles of good design.

Skills and knowledge matter in every medium, and jumping to an intermediate level has the potential to mislead novices about what they are really accomplishing.

Skipping skills may erode patience.

Skipping skill-building may also increase impatience with complex, difficult, long-term work. If people can sit down at a computer and manipulate complex music, where's the motivation to practice the violin?

Yet there seems to be great value in sticking with complex long-term projects like learning to play the violin.

Simulations have weaknesses.

Drawing programs, for all their powers, make it very difficult to produce a line with as much life as any child can create with a crayon. Even an extremely advanced and patient user of music software would have trouble producing a performance as nuanced as a child’s singing. Simulations have power because they’re not the real thing, and they also lose power because they’re not the real thing.

Every technology—not just learning technologies—is a mix of good news and bad news. Each technology offers potential advantages and potential pitfalls. Skillful use of technology has always included understanding both sides.

The results depend not just on the user’s skills but also on assumptions and intentions. A computer can’t guarantee outcomes any more than a pencil can. Computers, like all other technologies for learning, need to be used mindfully.

Part 3: The computer as a new medium

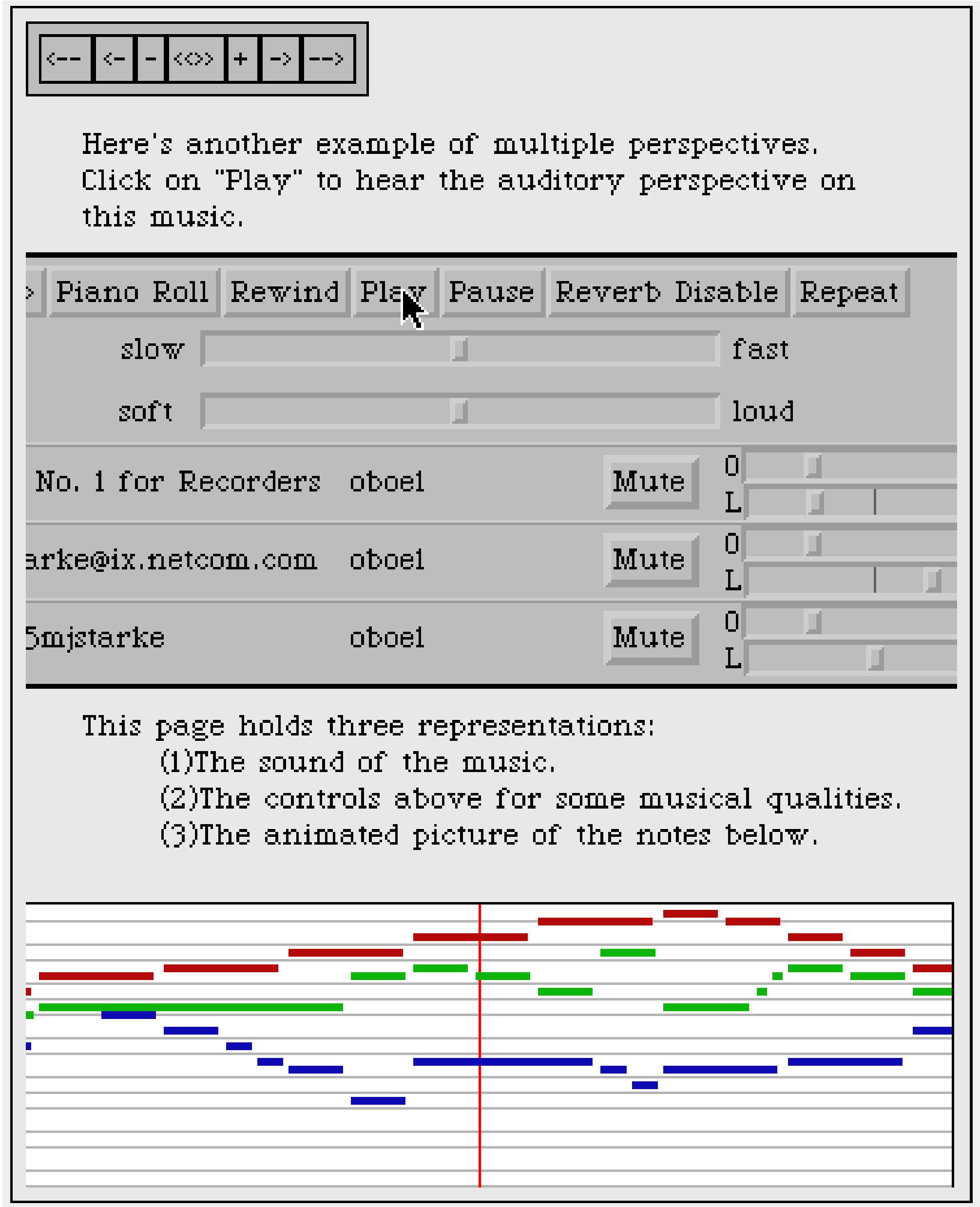
Computers are so new that their nature and potential are still unfolding, but the medium offers two unique characteristics: multiple perspectives and the capacity to model dynamic processes. Whatever new genres emerge in this medium will probably exploit those characteristics.

One way to develop a new medium is to create a “literature” of examples to show what the medium can do, to provide fodder for discussion, and to inspire improvements. In a new medium, even bad examples can be helpful, by showing what doesn’t work.

Some Squeak projects are examples that exploit and demonstrate the medium’s possibilities (and display its weaknesses). Through this work, Squeakers are developing two new genres that are unique to computers while constructing the software infrastructure for building and sharing a literature.

Multiple perspectives

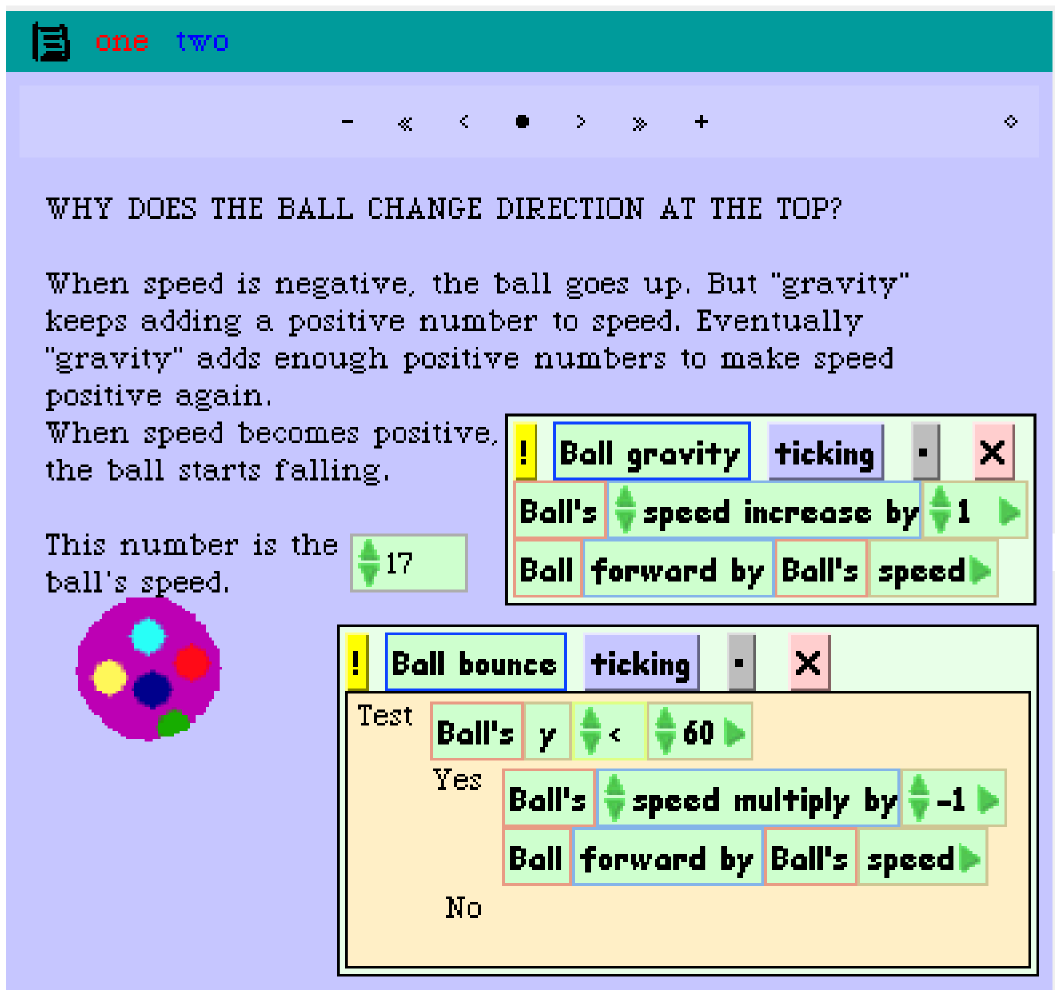
Because computers are so good at imitating other media, they can manifest a thing (image, concept, relationship, datum, etc.) in several different ways. Computers can present in different visual forms, and they can also address other senses, most often hearing, touch, and the kinesthetic sense. By offering different ways to examine and explore, computers can help a person understand in more than one way.



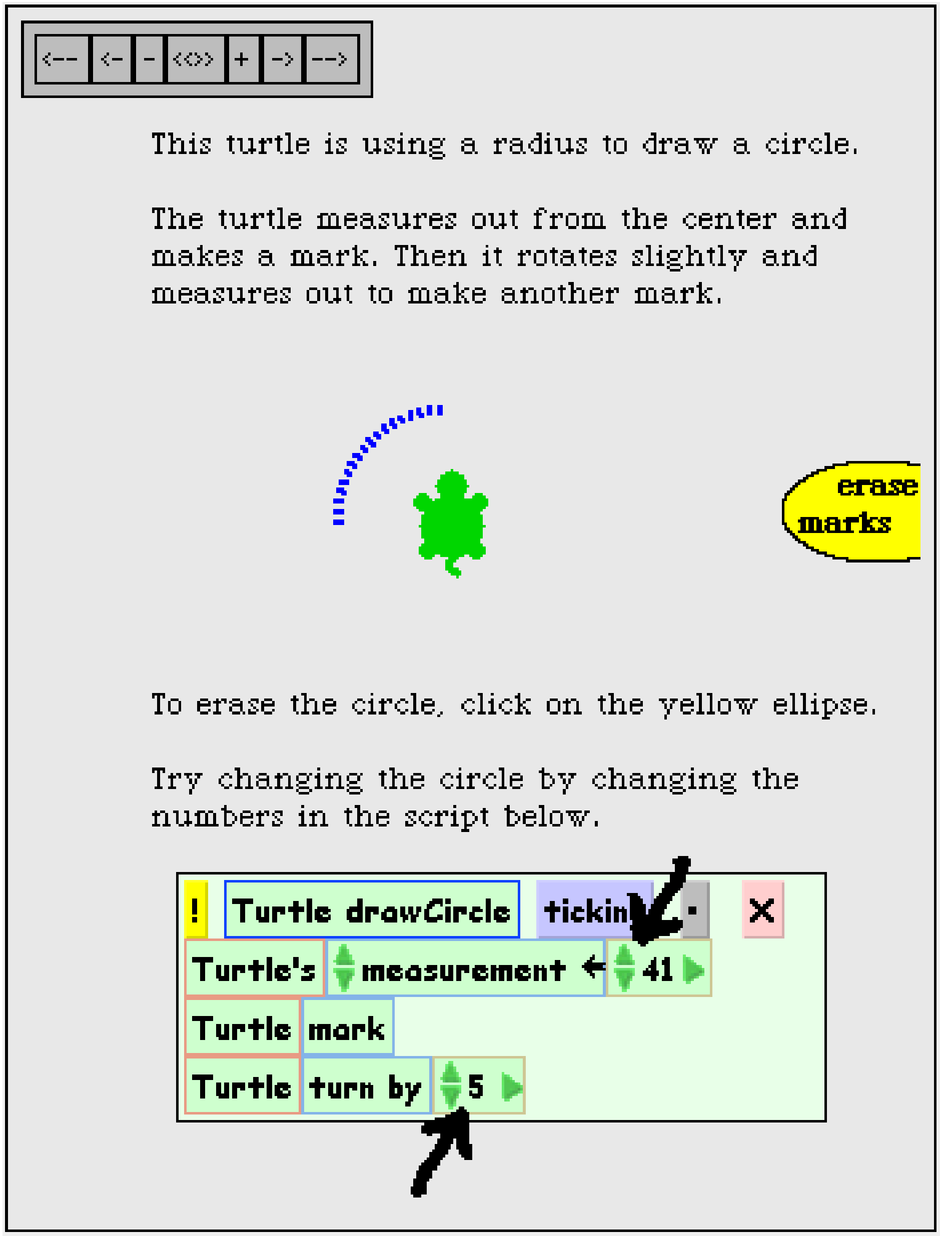
For learners, the computer's multiple perspectives offer opportunities to engage a topic in different ways. This means that the medium can address different learning styles. It also means that one learner can explore the same issue in a variety of ways—through a variety of senses, with different representations—and thus deepen the learning. With or without computers, good teachers use multiple perspectives to reach different learners and to deepen learning.

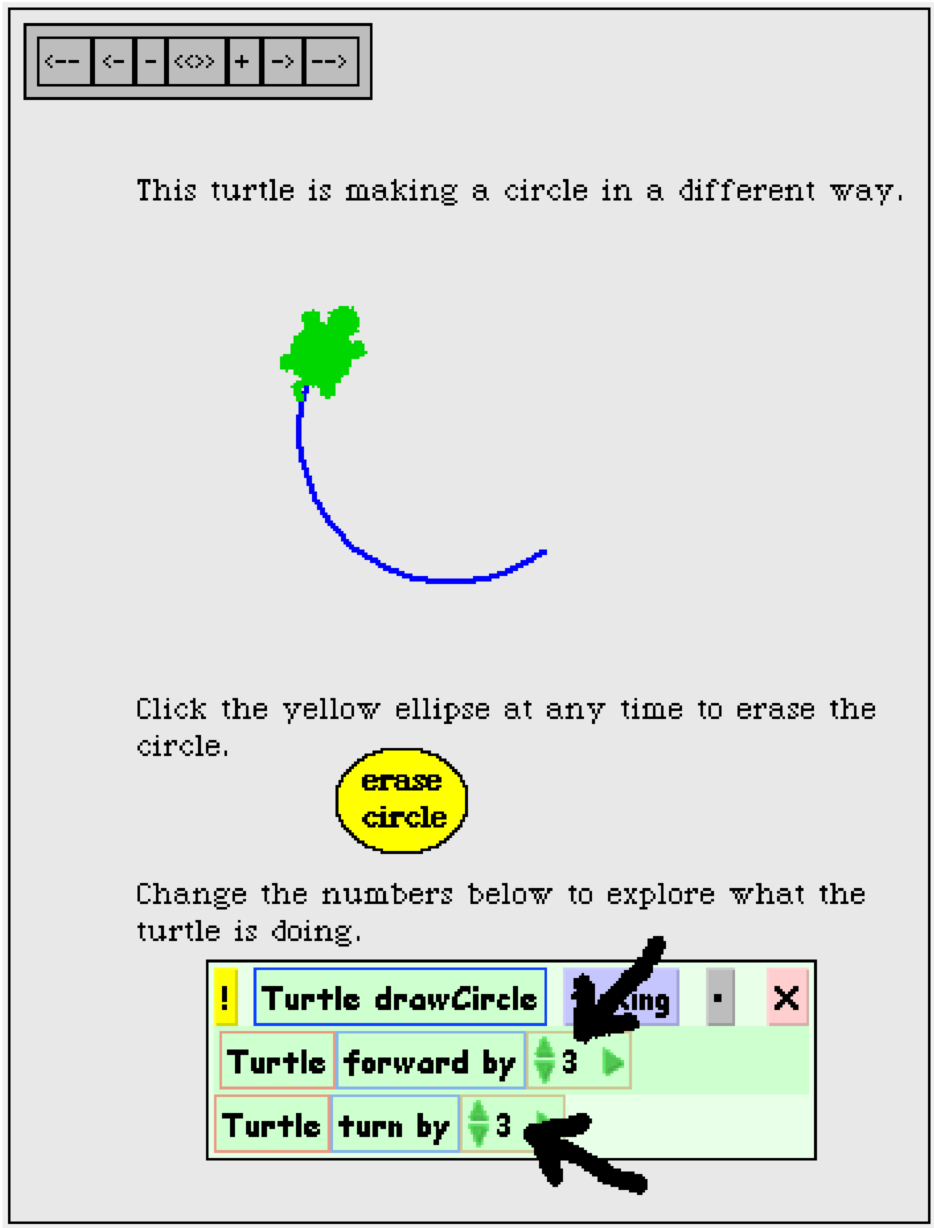
A Squeak-based lesson in gravity includes text, animation, and the program code controlling the animation. Users can change the animations to learn more about how to program a simulation of gravity. They make changes by manipulating the drawing, by clicking on buttons to alter the code, or by altering the code directly. The representation is textual, visual, and symbolic, and can be engaged with physically, visually, textually, or symbolically. (Such a lesson extends, but does not replace, physical and observation activities with real-world objects.)

The following illustration shows a perpetually bouncing ball.



The next pair of illustrations shows two ways to think about a circle. Working with these representations might foster new perspectives on circles.


<-- <- - <<>> + -> -->
This turtle is using a radius to draw a circle.
The turtle measures out from the center and makes a mark. Then it rotates slightly and measures out to make another mark.
erase marks
To erase the circle, click on the yellow ellipse.
Try changing the circle by changing the numbers in the script below.
! Turtle drawCircle tickin
Turtle's measurement ← 41
Turtle mark
Turtle turn by 5



Both scripts produce circles, yet they are made in different ways. By working with the two scripts, a person might begin to see that a circle can be understood in different ways, with or without reference to a center point.

Another perspective that might emerge from working with these two kinds of circles is that a circle can be seen as a process instead of a thing. Surprisingly, circles can emerge from different processes.

Of course working with multiple perspectives doesn't assure insight or understanding. Such work might even lead to negative outcomes such as reluctance or inability to maintain any point of view long enough for fruitful exploration.

Dynamic processes

Because they are so good at making changes, computers can facilitate learning about dynamic processes—processes in which change is ongoing. As a time-based medium, computers provide particularly helpful ways to understand processes in time.

Humans have developed many ways to capture changes in time, such as musical notation, mathematical symbols, stories, and plays. More recently, we have developed ways to store and replay time-based phenomena on film, tape, and disk. Other devices such as oscilloscopes and plotters represent oscillations, waveforms, and other fluctuations in time. Now, with computers, we have ways to link our notations with our stored replays. We can, for instance, link a film's shooting script with the film itself, with a storyboard, and with other ways of understanding the film. We can link a recording of music with the score, with analysis, with simulated performances that can be altered by the user, and so on. We can link oscilloscope-like pictures of waveforms with the equations that describe them. Changing one of the linked representations alters the others, too.

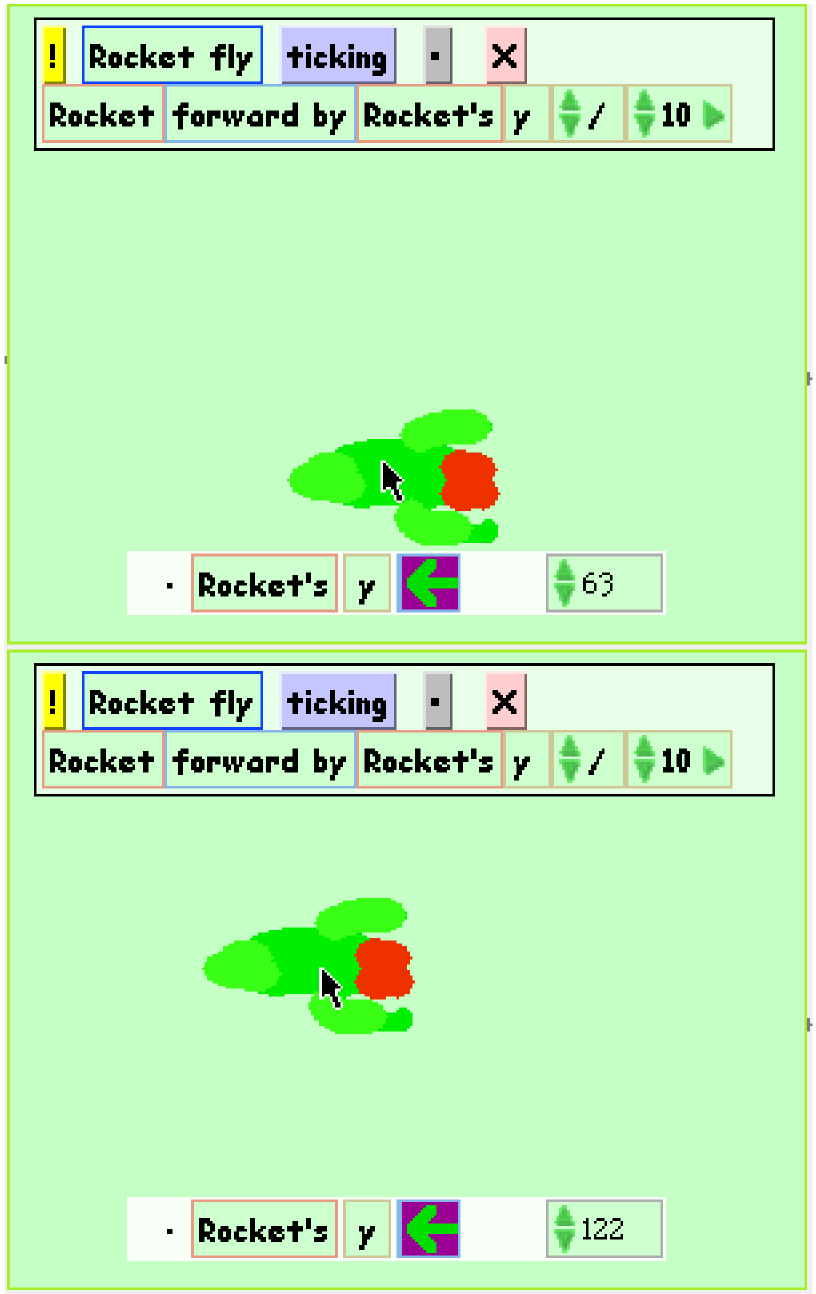
Naturally there are tradeoffs and risks in linking different kinds of representations. In general, notations derive their power by focussing on key phenomena or relationships and disregarding others. Linking notations directly to output must be done carefully, then, because a notation may leave out information or variables that are crucial to the liveliness and vividness of the output. Making the notation more complex may help, but it may also make the notation harder to understand.

For example, music software can link music notation with sound output. However, music notation is not a complete recipe for a performance. Because music notation does not include all of the details that make a performance vivid and soulful (human performers add those details as they perform), computer playback of music notation often sounds mechanical and unmusical. Using a computer to link musical notation to performance may limit the user, omitting crucial elements simply because they can't be notated. Some music software works around this problem by providing ways to add information to the score, but this makes the notation very complicated, dense with information, and difficult to read. Despite its limitations, computer playback of music can be very useful and even exciting, especially as a compositional tool.

Linking mathematical symbols to computer animations may help some learners understand mathematics. The math used to describe a motion formerly lay unmoving on the page, eluding the understanding of those who could not make the mental connection between the symbols and the movement. With a computer, the math can be implemented in software to run a simulation. Changes to the math in the program result in immediate changes to the simulation. Changes to the simulation affect the math.

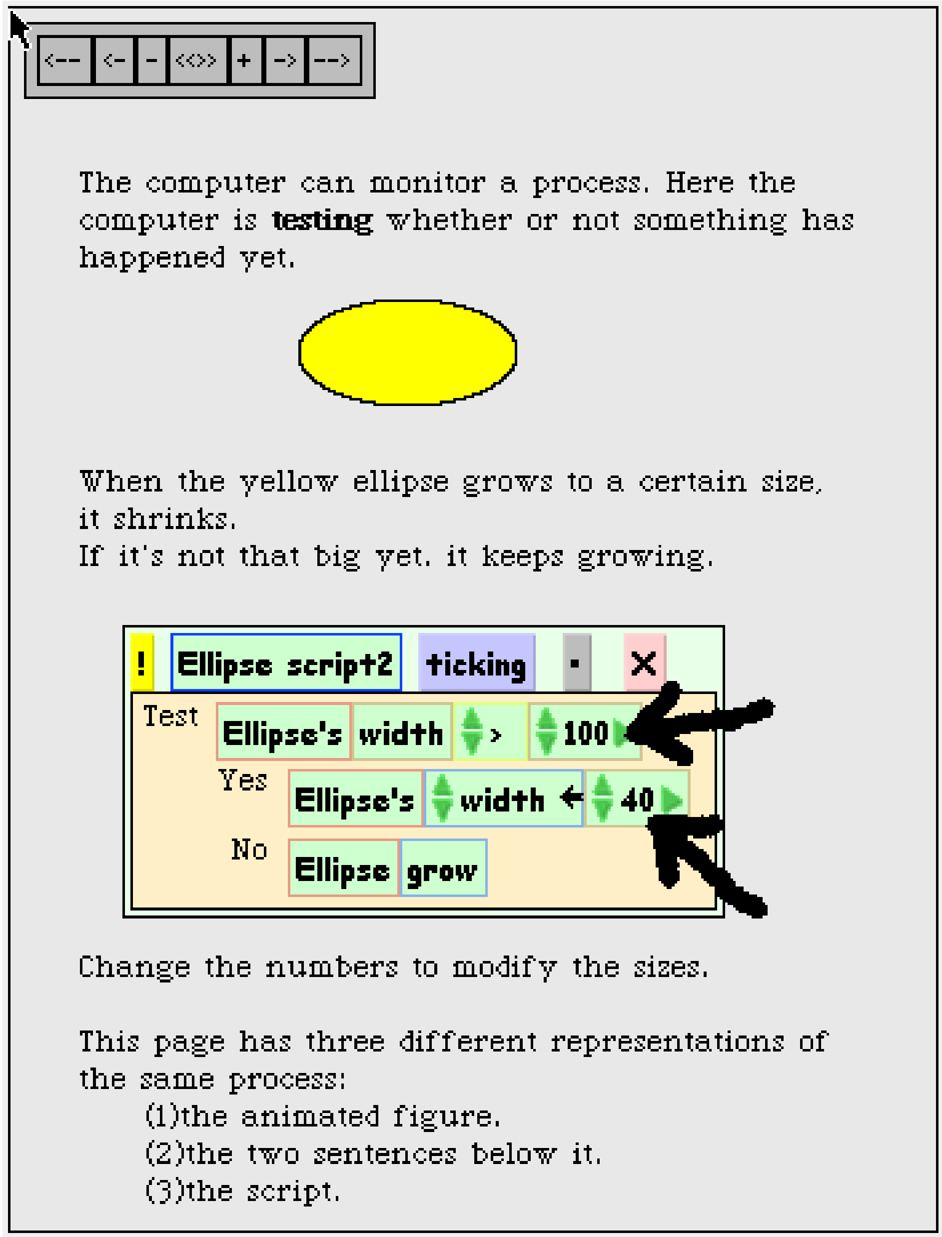
Making direct manipulations to a simulation program might help learners deepen their understanding of phenomena, principles, symbolic representations, and the relationships between them.

In the following simulation, the rocket's speed is dependent on its altitude. Moving the rocket higher or lower in the window raises or lowers its velocity. The user can move the rocket by dragging it or by changing the value of its y coordinate.



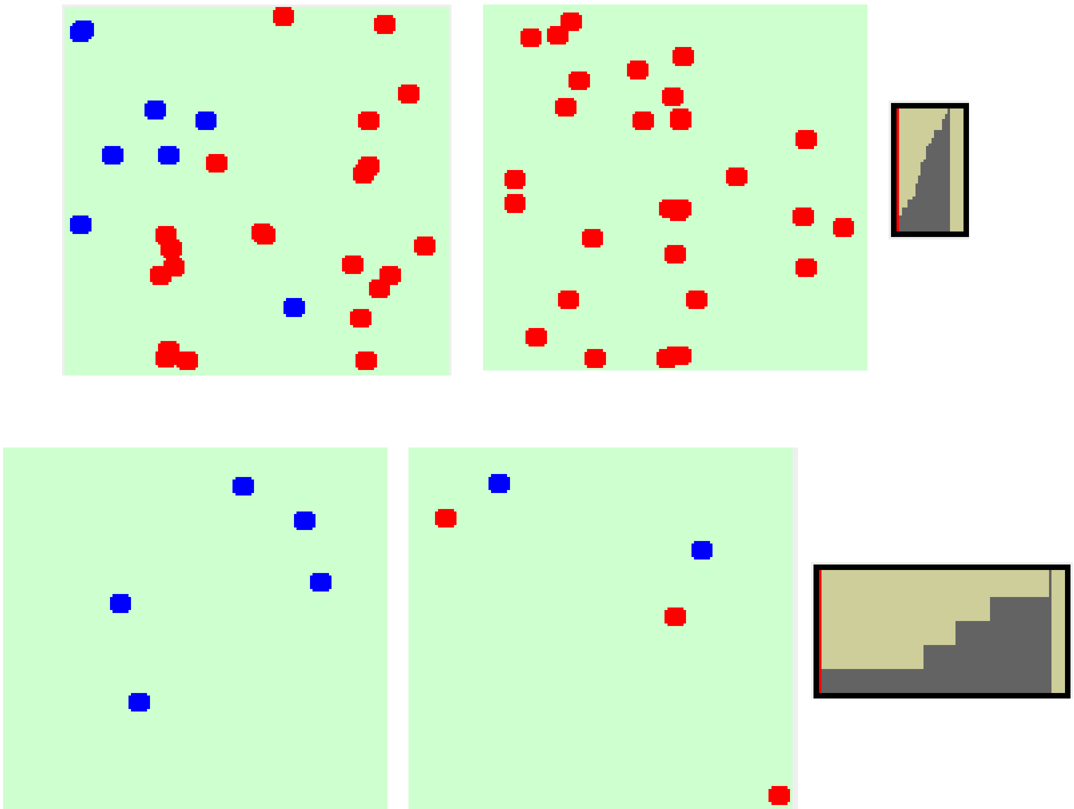
Note that moving the rocket changes its y-coordinate, and changing the y-coordinate moves the rocket. These changes affect the calculation that determines the rocket's velocity.

The following illustration shows another simple dynamic process. The yellow ellipse gradually expands to the right, until it reaches a certain size, when it collapses and begins growing again. (The original purpose of this example was to illustrate how computers can model dynamic processes, not to teach this particular process.)



Similar examples should help people to grasp higher-order fluctuations as well. For instance, if velocity increases by some rate, and that rate also increases over time (an accelerating acceleration), a computer simulation could allow a learner to manipulate both rates of increase and watch what happens to the simulation.

Among important dynamic processes are complex systems with many variables or many objects. Immensely helpful to professional scientists trying to understand complex systems, computers are beginning to help non-professionals understand them as well.[8] Here is the Squeak "bouncing atoms" demonstration, which shows a relatively simple simulation of a complex system. One of the careening dots can "infect" others with a new color when they collide. The simulation continues until all dots have acquired the new color. How long will it take for the whole population to become infected? Users can run the simulation with different conditions and generate graphs of the results.



Snapshots from an "infection" simulation. The simulation was run twice, with different size populations. Graphs show the rates of infection.

Of course playing with simulations does not guarantee any particular learning. That depends, as always, on the users' intent and attention.

Genres for a New Medium

Every new technology begins life by imitating older technologies. The first

[8] Richard Dawkins' "Blind Watchmaker" program was one of the first programs designed to simulate a complex system for lay people. By modeling the interaction of mutation and selection, the program provided hands-on experience with the forces underlying evolution. Part of the program's effectiveness was that it required the user to play an active role, that of selection forces.

piano music was rather like harpsichord music. Early films were rather like filmed plays. The first cars looked like carriages.

As people gain a better understanding of the unique characteristics of a new medium, new genres develop. Liszt wrote piano music that would have been unimaginable a couple of generations earlier. Films developed their own unique grammar. Cars began to take on a different look more suited to their nature.

Genres in the computer medium are still developing. The medium's creators are working to understand its potential, its grammar and syntax, its limitations and possibilities. The medium's unprecedented flexibility and unceasing technical transformation may make it difficult for stable genres to emerge, but just as humans eventually figured out what film was all about and what pianos were good for, we will eventually develop genres unique to the computer medium. (And, just as piano music, films, and cars have continued to evolve, computer genres, once they appear, will continue to develop.)

Some transitional genres have already appeared. Dynamic spreadsheets are a new twist on an old genre. Desktop publishing is a new way to create older genres. Websites, on the other hand, are something new and different, probably a transitional form on the way to becoming a new genre. Video games provide examples of new genres.

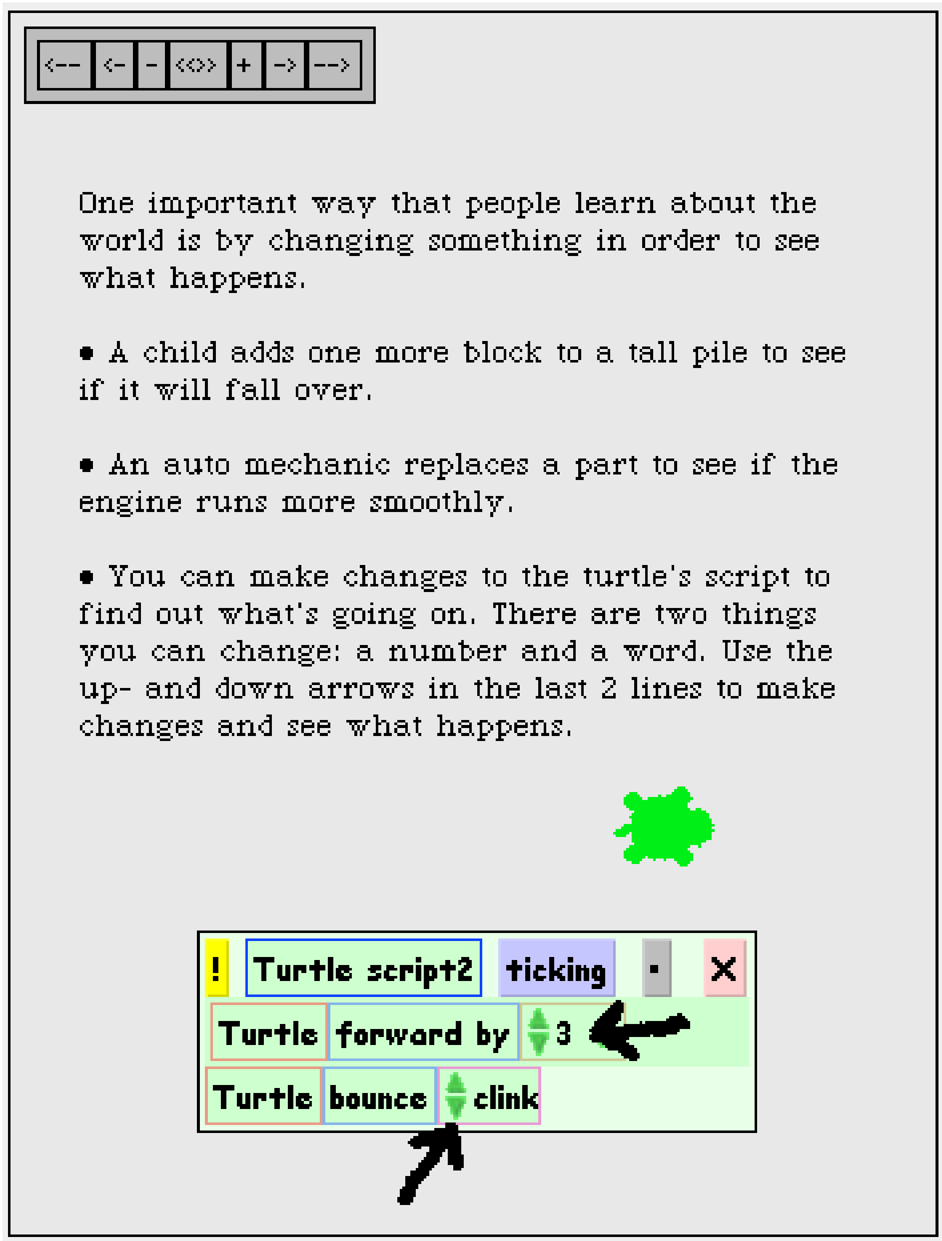
New genres in the computer medium will certainly include different kinds of learning environments. The Squeak community is working with two new genres for learning: the Active Essay and the SqueakToy.

Active Essay

An "Active Essay" is a new kind of document combining words, simulations, and programs.[9] The user works directly with multiple ways of representing the concepts under discussion. By "playing with" the simulations and code, the user gets some hands-on experience with the topic.

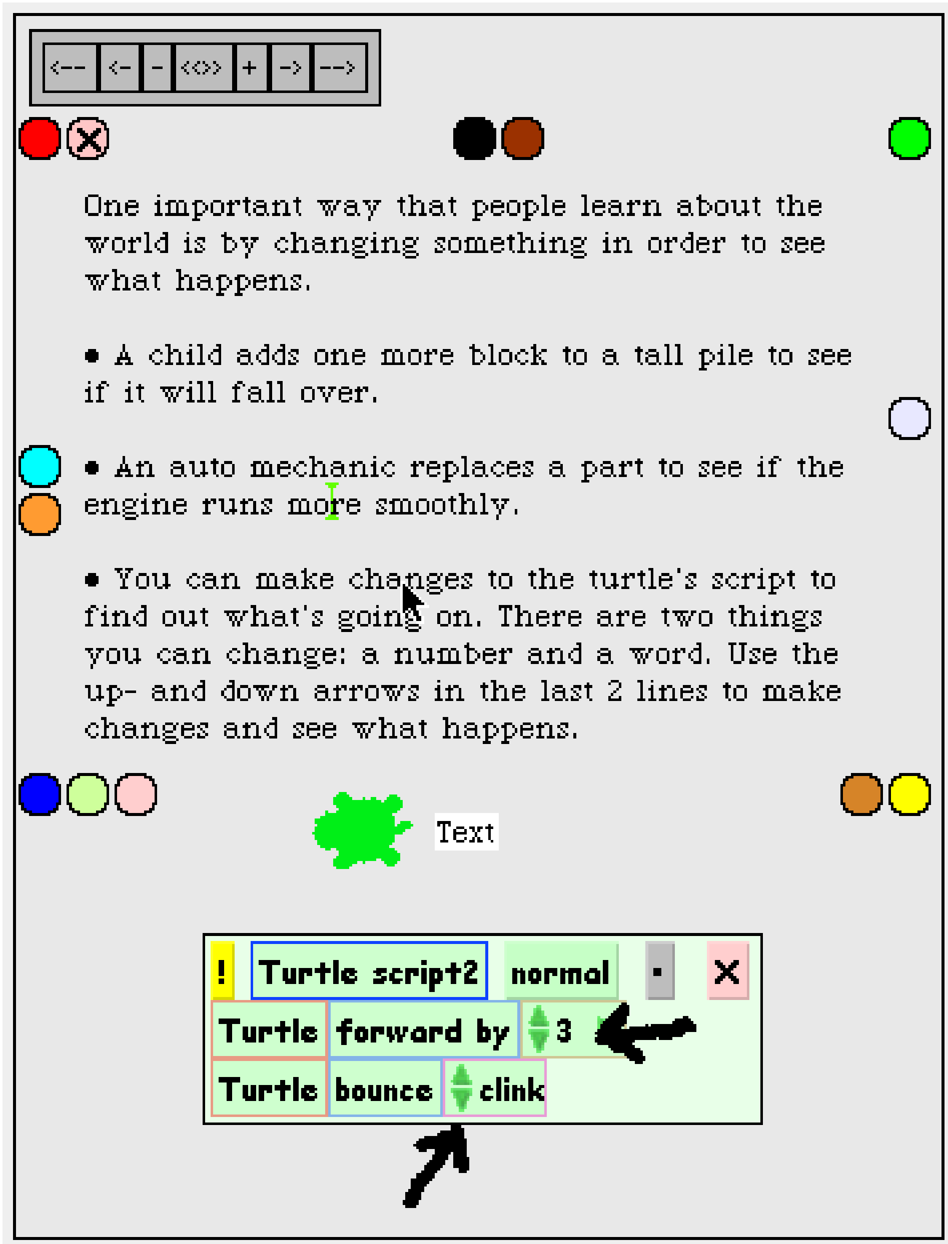
The next illustration, like many in this chapter, is a "page" from an Active Essay about the computer medium. When the page first appears, the turtle is moving.

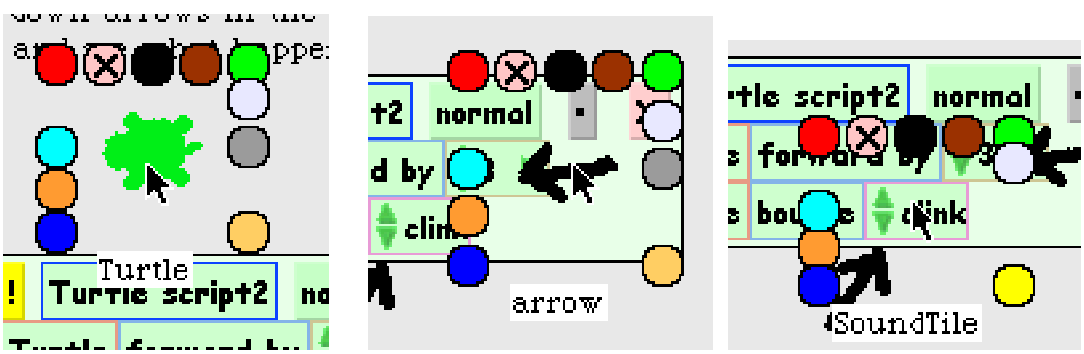
[9] Alan Kay and Ted Kaehler have developed the Active Essay, and have constructed some examples. See Ted's web page, www.webpage.com/~kaehler2/, for more information on Active Essays.



Squeak’s design makes it easy to display functioning code alongside other text, graphics, animations, and simulations, so that the code and its outcome can be observed simultaneously. A window can contain any kind of object, and endusers can examine any object and the code that runs it. Learners who know the Morphic interface can examine objects in an Active Essay to find their names and scripts, and can easily redesign objects and their programs.

In the following illustrations, a user selects different items on the screen: a block of text, graphic elements (a turtle, an arrow), and a scripting tile. Once selected, any of these objects can be revised, repainted, rewritten, relocated, or rescripted.

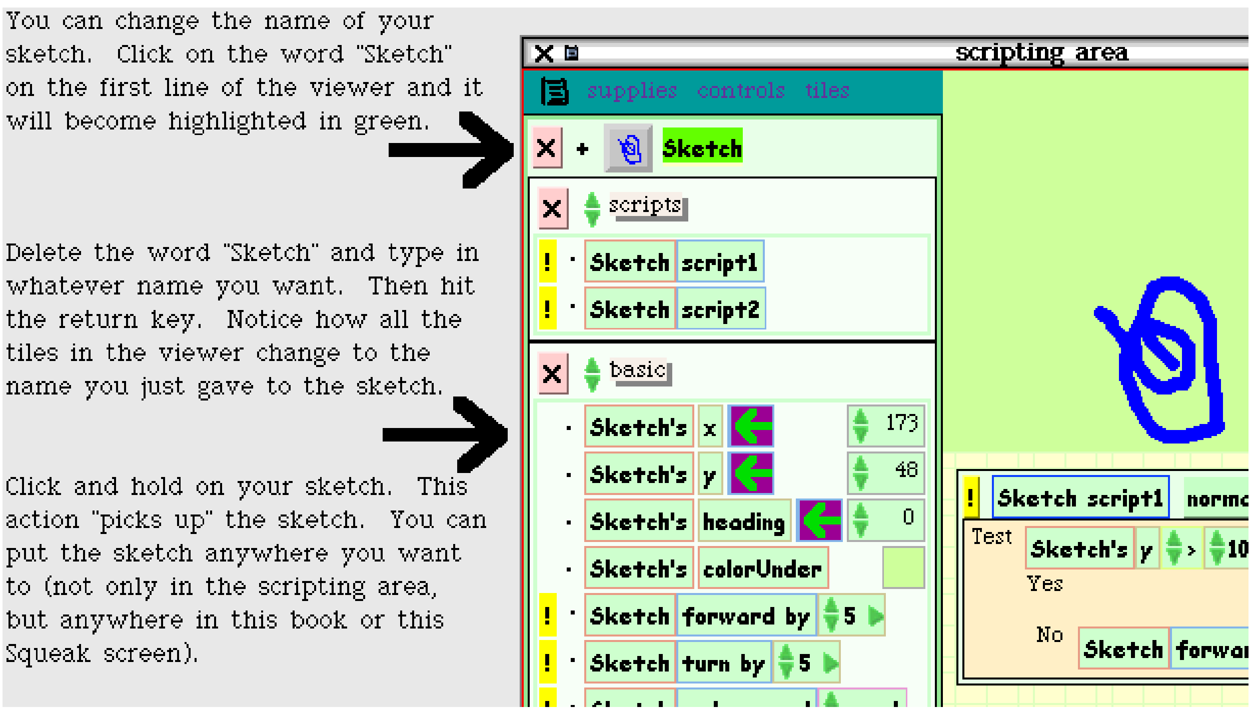

<--
<-
-
<<>>
+
->
-->
One important way that people learn about the world is by changing something in order to see what happens.
• A child adds one more block to a tall pile to see if it will fall over.
• An auto mechanic replaces a part to see if the engine runs more smoothly.
• You can make changes to the turtle's script to find out what's going on. There are two things you can change: a number and a word. Use the up- and down arrows in the last 2 lines to make changes and see what happens.
Text
!
Turtle script2
normal
Turtle
forward by
3
Turtle
bounce
clink


Turtle
Turtle script2
normal
arrow
SoundTile

Perhaps the main advantage of the Active Essay is that the user is not just a reader of text or an observer of outcomes, but a participant in the activity. The user can be invited to rewrite text, design graphics, write code, or change existing programs. An Active Essay is not just a collection of information and experiences, but a laboratory for experimentation and exploration.

One potential use of the Active Essay is in teaching about Squeak. There is a tantalizing possibility that the computer might replace printed manuals and become the medium for learning about itself, about software, and about programming.

Here is part of a page from an Active Essay by Naala Brewer,[10] using Morphic to teach about viewers in the Morphic interface.



Much remains to be learned about Active Essays, but here are a few observations about strategies and style in this genre:

• The learning that takes place flows from the user’s actions and thinking. A good Active Essay would stimulate thinking through appropriate action and exploration. An Active Essay is less a place to display an author’s cleverness than a way to call forth cleverness from the user.

• Computer screens are still not good places to read text. At this point

[10] Naala Brewer constructed a set of lessons about the interface and projects to build while she was a research intern at Disney, working with the Squeak team. Contact her at Naala.Brewer@disney.com.

the most exciting thing about Active Essays is not the text, but the participatory activities.

• Fitting everything onto the screen can be a problem. As the medium develops, we will better understand what needs to be visible, what can be hidden away, and where to store hidden elements.

• An ongoing problem is how to carry the user through a sequence of actions and make sure that the user did them all correctly. One strategy is to invite the user to make some changes, and then to continue on a new "page" with all of those changes made correctly.[11]

• Instructions and guidance may be given in many ways: in written form, by recorded voice, by animated character, by animated example (i.e., numbers move to show you how to move the numbers), by video clip. Although early prototypes are rather like books with text and illustrations, the Active Essay will probably develop into something quite different from a book.

• Some Active Essays may need to have intelligence, or at least a capacity to track the user's doings in order to suggest next steps.

• Users' levels of experience might determine how they interact with an Active Essay. Novice users would mainly participate in readymade activities, while more experienced users might want to "look under the hood" to find out why objects behave as they do, and advanced users might want to learn more about how a simulation or the essay itself is constructed.

SqueakToys

While Active Essays dispense information and experiences, SqueakToys[12] are construction projects or kits. Experiences and information are involved, but the center of the activity is making something.

Background: Learning by Making Things

Although much of what people need and want to learn appears to be informational, the most challenging part of education is to connect new skills and information with what the learner already knows. If new learning is not connected in this way, the learner may not be able to retain it in any usable form. (For example, if education were simply a matter of dispensing information, then the barrage of data about environmental crises would have been sufficient to change Americans' patterns of consumption.) Most learners need help integrating new learning. To facilitate such integration is not trivial, because every learner arrives with a unique set of assumptions, obstacles, previous experiences, skills, motivations, and knowledge.

One great way to learn and to integrate learning is to make

---

[11] Thanks to Alan Kay for this insight; he noticed the strategy implemented in a tutorial for the software Alice.

[12] The working name for such projects, "eToys," has been scrapped due to trademark conflicts, and a new moniker is needed. I'm calling them "SqueakToys" until we think of something better.

something.[13] Whether the something is a sculpture, a house, a musical composition, a performance, a theory, a computer program, or an essay, making it is a way to find out what you know and what you don't know, to provide a context for new learning, to integrate that new learning, and to give form to what you have learned. In an educational setting, the finished product also provides clues about what the maker knows and doesn't know—especially about what knowledge and skills the maker is able to put to use. Educators are increasingly, and appropriately, concerned that learners be able to activate and use what they learn.

When I write an essay, the act of writing helps me—forces me—to see connections between things I know and to discover ways to organize the information and the connections. Writing may help me remember important items I'd forgotten. I may also have to gather new information, and because I need that information for my essay, I am likely to integrate it with what I already know.

A well-designed educational project makes learning exciting and fun precisely because it requires the learner to use new information, skills, or understanding in some meaningful way. The learner must grasp the new learning well enough to assess how it relates to the project, and then make decisions about whether and how to use it. Sometimes this decision-making and learning happens below the threshold of awareness while the learner's attention is focussed on completing the project. Making something is one powerful way to initiate fruitful connections between what a learner already knows and what she is learning.

I also suspect that people learn more easily after activating what they already know. When making something puts a learner's existing knowledge and skills into action, potential connection points become "alive," ready to link with new learning.

Because learners vary in their ability to integrate new learning and in their learning styles, there is no one right way to teach. Learners need flexible facilitation approaches that can help each learner's unique situation. Making things can be one such approach; a well-designed project to make something can provide lots of entry points, varied learning opportunities, and multiple motivations for engaging with the project.

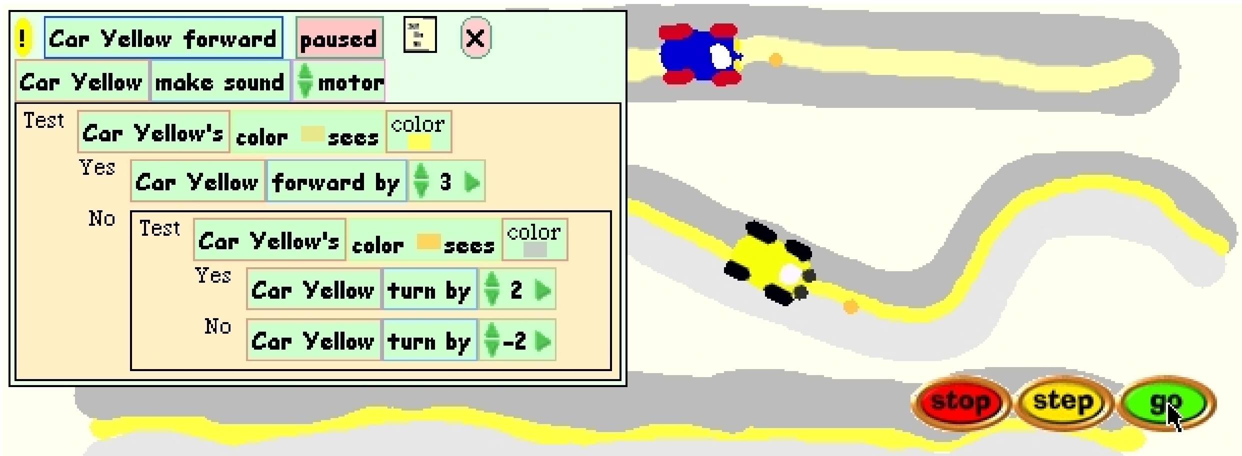
Learning by Making SqueakToys

One purpose of Squeak's Morphic interface is for non-expert adult programmers to design SqueakToys, projects for kids to build in the computer environment. Through making simple animations and simulations, children encounter ideas from mathematics, science, and programming and practice using them. SqueakToys are meant to be fun and fairly easy to build, capable of being mocked up spontaneously, without too much forethought—rather like sketching an idea on the back of an envelope.

---

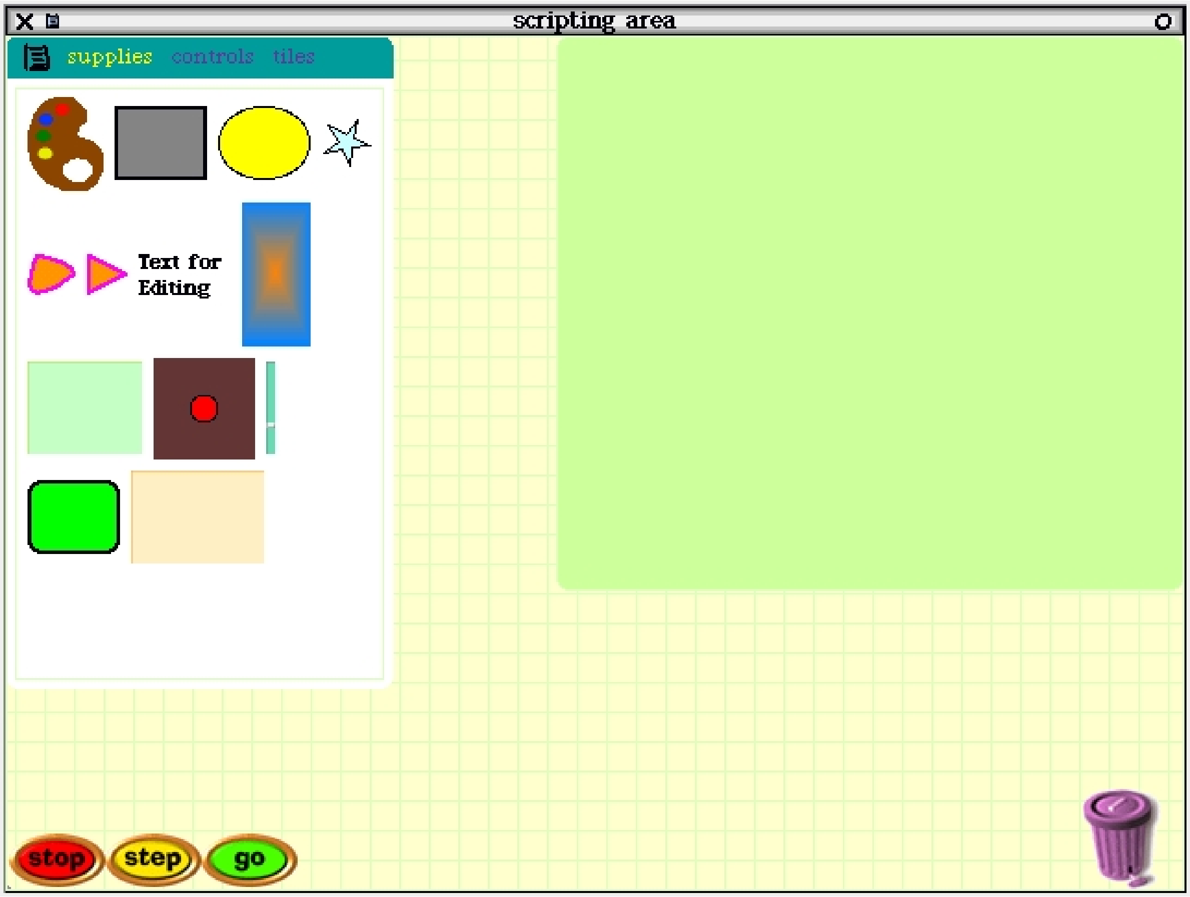
[13] I am indebted to Prof. Doreen Nelson of Cal Poly, Pomona, for her ideas about how constructing things—particularly objects in 3 dimensions—helps learning. Similar ideas from Seymour Papert and others influenced projects in computation and learning at the M.I.T. Media Lab, and such projects have now spread to other places through the efforts of Mitchel Resnick, Uri Wilensky, Yasmin Kafai, Amy Bruckman, and others.

SqueakToys are meant to be easy to modify and extend. In other words, SqueakToys should be easy to change as their maker's ideas develop.



"Robot Car" project designed by B.J. Allen-Conn, a teacher in the Los Angeles Unified School District. The cars automatically follow a curving road. The program for the lower car is displayed. Without seeing this script, several of the students were able to program their own simulations to exhibit the same behavior.

Combining features of previous systems that Squeak's designers admired, Morphic includes a simple painting interface and a tiling system for creating scripts for objects. In a basic SqueakToy project, the child draws an object or character and then gives it movement or other behavior by assembling a script from the tiles provided. To create such a project for children, an adult must build it and then figure out how to make the materials and concepts available for a child.



SqueakToy construction zone.

The SqueakToy endeavor is a multi-level environment for learning. In addition to whatever the children learn from building SqueakToys, adults observe how the children use the projects and learn from them. Meanwhile, the adults who design SqueakToy projects learn about programming and about what makes those projects effective. Squeak programmers are learning how to design an interface to facilitate the creative work of adult intermediate programmers, the construction work of children, and the support work of parents and teachers who help their children build SqueakToys. That’s a tall order for one interface, so there is much to learn.

The Disney Squeak team has a long-standing interest in designing systems to help novices learn programming. Squeak, Morphic, and SqueakToys are the latest fruits of that interest, and they are proving to be quite useful, both for the novices and for those attempting to understand how to help them. Squeak’s combination of power and flexibility has facilitated prototyping and redesign of interfaces and projects. Facilities for sharing projects and updates make it relatively easy to turn critiques of the system into improvements.

The interface has been sufficient for preliminary experiments with a few parents, teachers, and children building small-scale simulations and animations. For her classroom of fourth- and fifth-graders at the Open Charter School in Los Angeles, B.J. Allen-Conn designed several

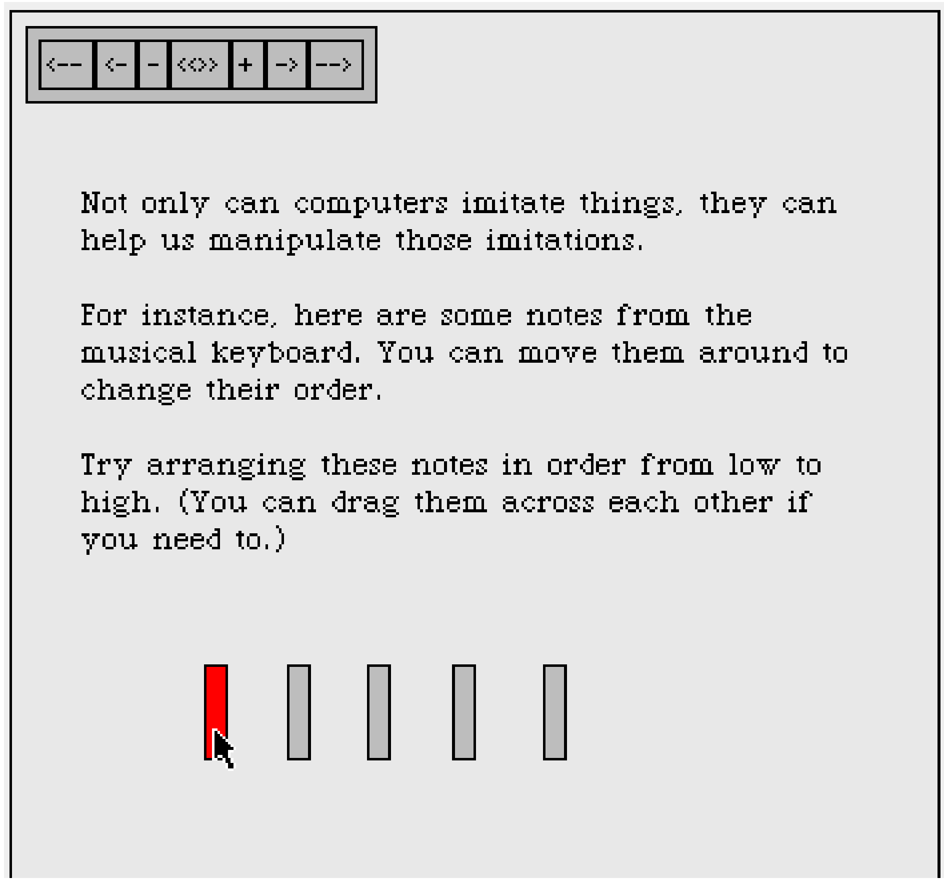
SqueakToy projects, and at first the children seemed chiefly occupied with learning the interface. Over the several weeks of their work with SqueakToys, several of the children became fluent enough to switch their attention to the mathematical thinking embedded in the projects. Some students, with help from the adults on hand, conceived additions to or variations on the projects and figured out how to implement their ideas.

Students absorbed in their work on a SqueakToy. Part of this assignment is to answer questions on paper. Open Charter School, Los Angeles.

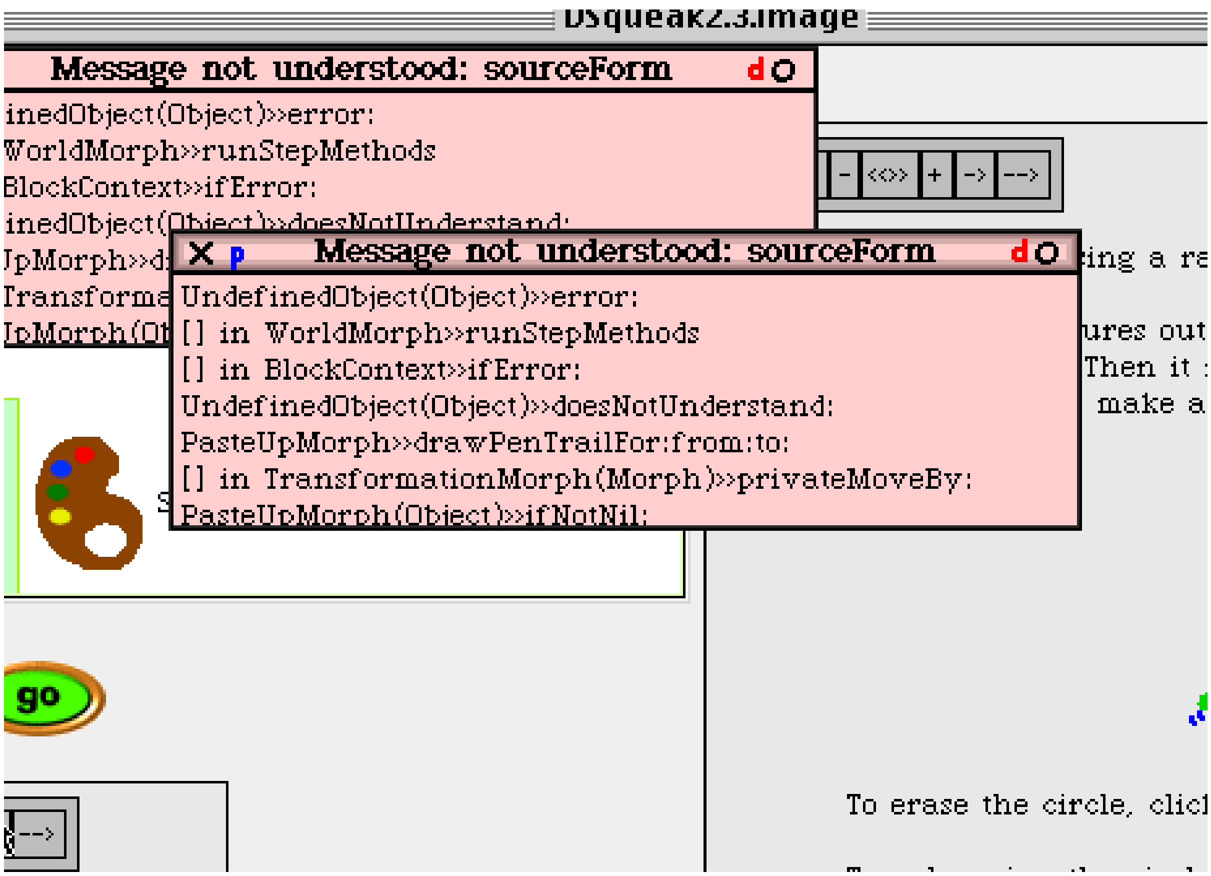
In fact, the children had so much fun working with Squeak that they frequently had trouble stopping their tinkering long enough to think about their SqueakToy's behavior. Computers can be so compelling, and making changes to them can be so much fun, that users become distracted from the very thinking a project is supposed to foster.

I have also found Morphic useful for constructing Active Essays aimed at adults. Its ability to construct and modify projects on the fly is particularly helpful and enjoyable.

Although much work of simplification and clarification remains, the Morphic interface offers some helpful ways of thinking about what non-expert users might need in order to do some programming, and Squeak makes it possible to implement those ideas so that they can be tested.



Among the remaining challenges, one of the greatest is to make it easier for users to share their projects between machines. Although Squeak works across platforms, the problem of conforming with different versions of a system subject to constant updates has not yet been solved for inexperienced users. Too often error messages are generated which baffle intermediates and beginners because they come from outside Morphic and look different from any other SqueakToy messages.



Some other difficulties are inherent to user-interface design, with its constant tradeoffs between conflicting needs. Building SqueakToys requires seeing both the simulation (or animation) and the code that runs it, along with instructions. How to fit everything onto the screen—particularly when it needs to be able to run on any platform—will be an ongoing problem with no single correct solution.

Because reading text from the screen is so unappealing—and early experiments confirm that children don't like to read SqueakToy instructions[14]—SqueakToy designers will need to discover other ways to communicate with users.

All these experiences and observations can contribute to better understanding the computer's potential as a learning environment.

[14] In Naala Brewer's experiment at a Saturday drop-in computer class, children enjoyed building SqueakToys but preferred adult coaching over reading the onscreen documentation.

Parting shots

Useful questions

Here are a few questions to keep in mind when designing computer-based learning environments:

• What is the person learning? How much of the learning is about how to use the computer or the software, and how much is about something else?

• To what extent does the project require thinking and/or imaginative response?

• How many possible right answers are there?

• How much opportunity for design choice does the user have?

• What is the intent of the project from the standpoint of the designer? From the standpoint of the user?

• To complete the project, what will the user need to discover? What will the user need to be given?

• How will the user's attention be drawn to thinking about the phenomena onscreen instead of just making changes or proceeding by trial and error?

• Who's having the fun?

Squeak learning communities

Just as Squeak has attracted a far-flung community of users and developers, SqueakToys and Active Essays are beginning to attract a community of project designers and teachers. At this writing, the community is tiny. As Morphic's interface improves, that community will probably grow. Another potential community is in homes, where children and parents may work with SqueakToys downloaded from websites. Eventually a community of children may share their SqueakToy projects via the internet. For the present, though, Morphic remains a promising and stimulating research system, serving users who have access to support from the system's designers.

For SqueakToys to succeed with a broad cross-section of children (not just with those with gifts and inclinations for programming), and particularly for SqueakToys to reach homes successfully via the internet, Squeakers will need to discover ways to help non-expert adults, such as parents, to facilitate children's use of the projects. Many adults do not know how to coach children's learning by asking questions or directing attention instead of offering answers. Internet-based SqueakToys will need to help the adults help the children.

In classrooms, computer-based projects in older languages like LOGO and HyperCard have stimulated collaborations between children. Because making something with a computer requires a variety of skills, children with different abilities and different learning styles can help each other. Some teachers report that computer projects also foster a more collaborative spirit between teachers and students, because teachers who are computer novices become learners alongside their students.

B.J. Allen-Conn and students discuss a SqueakToy project. Open Charter School, Los Angeles.

In comparison with older languages, Morphic offers an environment in which novices can try ideas with a relatively little pre-planning. The tiles and their displays reduce the need to remember names of primitives and what they do, so the system can be learned partly through experimentation. By fostering exploration and improvisation, Morphic may also foster collaboration.

Active Essays and SqueakToys are conceived not just for solitary, but also for children or grownups working together in small teams, helping each other to understand, solve problems, and develop solutions. Collaboration seems to suit computer-based construction activities, whether the collaboration is between children gathered around a terminal or between Squeak developers scattered around the world.