Logo*Update* The Logo Foundation Newsletter ▲ Volume 4, Number 2 – Winter 1996

In This Issue

The current fad in educational technology is multimedia. What does this have to do with Logo? In this issue of *Logo Update* we answer the question from two points of view. First, in "Logo and Hypermedia" beginning on this page, I describe the evolution of Logo into a hypermedia authoring environment. The leading edge of this process is the newly released MicroWorlds 2.0 from LCSI. Second, in "HyperLogo: Logo Like You've Never Seen It!" Mike Westerfield describes how Logo works within HyperStudio, one of the most popular and widely used multimedia programs available today.

Mike and I express very different points of view about Logo and multimedia. Underlying our debate is a more fundamental controversy about how programming, sometimes referred to as "scripting", should be used within computer applications. But we agree in general that software should be programmable, and specifically that Logo is a good choice for multimedia environments.

If you'd like to explore Logo and multimedia more deeply, join us for one of several Logo Summer Institutes that will take place during June, July, and August in Maine, Minnesota, and New York City. These institutes also offer the opportunity to work with several other Logo strands including turtle geometry, simulations, and game design. Look at pages 4 and 7 for more information.

Another summer activity coming up is the third annual Logosium, which will be hosted by the renowned St. Paul Logo project in St. Paul on June 14, the day after NECC. Look at the Call for Partici-

continues on the next page

Logo and Hypermedia

by Michael Tempel

At a workshop about a dozen years ago Robert Knight of the Lubbock, Texas Public Schools showed his Logo multimedia creation. Written in Terrapin Logo for the Apple II, it told a tale of a frog in a lily pond. The triangular turtle not only drew the scenery but was also an actor in the story. The narration, complete with musical accompaniment, came from a cassette tape recorder sitting next to the Apple.

We've come a long way since then. With the advent of MicroWorlds, Logo has evolved into a full-featured hypermedia authoring program. And coming from the multimedia side, HyperStudio has incorporated Logo as its resident programming language. (Look at the article by Mike Westerfield, which begins on page 8.)

The use of Logo for creating multimedia projects received a boost back in 1986 with the introduction of LogoWriter. Several features of LogoWriter combined to make it especially suitable for storytelling and reporting. Text could be written directly on the screen and edited as with any other word processor. Carrying on a tradition developed in earlier versions of Logo, the four LogoWriter turtles could take on a variety of ready-made or user-created shapes and move around the screen as birds, dancers, or space ships. Turtle shapes could also be "stamped" to add to a background drawn using turtle graphics.

But there's more to multimedia than mixing media. For thousands of years people saved data and told stories in scrolls, friezes, and books, all of which forced a linear organization. Electronic storage frees us to read and write in nonlinear ways. Point at a word or picture and you jump to another part of the document. The path you follow through a document may be different today than it was yesterday, and someone else will follow yet another route. The software can remember where you've been so you can backtrack.

This approach, familiar to anyone who has used HyperCard, a CDROM encyclopedia, or the World Wide Web, was originally known as "hypertext" when computer documents were only text. It evolved into "hypermedia" as graphics, sound, and moving images were added. The term "multimedia" lacks the important reference to hyperlinking, but has become commonly used to refer to hypermedia environments.

Two features of LogoWriter make it a natural hypermedia environment. First, a Logo project is organized as a collection of pages. In this electronic scrapbook the order of the pages doesn't matter. You can go from one to another via any path you choose. Second, text on the LogoWriter screen is "active." As in any word processor, you can select an area of text and then cut or copy it. But LogoWriter goes beyond this. The primitive **selected** reports the selected text. If the word you select is the name of a LogoWriter page, then the command **getpage selected** takes you to that page. Thus, pages may be linked directly to words in a body of text. With a little programming, you can create a system in which putting the cursor on a word and pressing Control-Something sends you off to another page.

Andy David wrote such a program almost as soon as LogoWriter hit the street. Over the next several years Eadie Adamson and I developed similar programs, which we used with her fourth and fifth graders at the Dalton School in New York City. Eventually our work was published in 1991 by LCSI as *LogoWriter Hypermedia Tools* for use with the Apple II version of LogoWriter. But by then the world was moving on to the Macintosh / Windows style of interface, and LogoWriter was not.

continues on the next page

Logo Foundation
250 West 57th Street
New York, NY 10107-2228
Telephone: 212 765-4918
FAX: 212 765-4789
email: michaelt@media.mit.edu

The Logo Foundation is a nonprofit educational organization incorporated in New York State.

Logo *Update* is published three times yearly by the Logo Foundation. Subscription is free.

In This Issue

continued from page 1
pation on page 20.

Finally, you'll be pleased to know that the Logo Foundation has a new policy of offering discounts on purchases of all commercial versions of Logo that we distribute. This includes all the products advertised in this issue of *Logo Update*. So if you find something that interests you, turn to page 18, check out the discount price, and order from the Logo Foundation. Even bigger discounts are available when you purchase software in conjunction with one of our workshops or Summer Institutes. Contact us for details.

Michael Tempel

Logo and Hypermedia

continued from page 1

MicroWorlds

A long period of stagnation in Logo development ended abruptly in 1994 with the introduction of MicroWorlds. The overall organization of a project as a collection of pages was carried over from LogoWriter. But a long list of changes, large and small, added up to a quantum leap forward for Logo. Many of the improvements are especially helpful when one is using MicroWorlds for hypermedia creations. Backgrounds can be created using drawing tools and imported images. Text can be more flexibly formatted and positioned on the screen than was possible in LogoWriter. You can have as many turtles as memory allows. They can grow and shrink and carry multicolored shapes. Buttons can link pages and run programs.

Is MicroWorlds a "hypermedia Logo"? As with most Logo environments, people are using it for many purposes. Becuase of its object orientation and multitasking capability MicroWorlds Logo is ideal for game-construction. It's also a good laboratory for building simulations in physics, biology, and ecology. But one of its greatest strengths is in supporting hypermedia projects, and a just released upgrade, MicroWorlds 2.0, further enhances the hypermedia capabilities of this Logo environment. Now you can play audio CDs and video laserdiscs under program control. QuickTime™ movies can be incorporated into MicroWorlds projects. You can choose from a number of fancy transitions between pages. A "full screen" mode blanks out the command center, tools palette, and menu bar leaving only your creation visible. A MicroWorlds Player allows you to distribute finished projects to people who do not have MicroWorlds, just as with HyperCard and HyperStudio.

HyperStudio

HyperStudio is a hypermedia authoring program that enables the creation of projects incorporating text, graphics, sound, movies, audio recording, and laserdisc segments. It includes Logo as its resident programming language. Roger Wagner, who developed HyperStudio, sees it as an answer to the question: "What will the word processor be in a world of multimedia?" HyperStudio was originally developed for the Apple IIgs and then for the Macintosh. A Windows version has now been released.

Are MicroWorlds and HyperStudio pretty much the same – hypermedia applications that include Logo? I think not. Sharon Yoder has written *Introduction to MicroWords – A Logo-based Hypermedia Environment*. The key term is "Logo-based," which is an accurate characterization of MicroWorlds, but not HyperStudio.

Before I go any further with this comparison, maybe some more information about HyperStudio is in order. I think that Logo *Update* readers are at least somewhat familiar with MicroWorlds, if not from direct experience, at least by way of articles in past issues of this newsletter. This is probably not the case with HyperStudio. Rather than tell you more myself, I asked Mike Westerfield to write about HyperLogo and its place in the HyperStudio environment. He should know best. He wrote HyperLogo. If you haven't read "HyperLogo: Logo Like You've Never Seen It!" I suggest that you take a break from this article, turn to page 8, and go read Mike's piece now.

Welcome back. HyperStudio with HyperLogo sounds good. So as Mike asks, "Where have all the Logo programmers gone?" Why haven't we all jumped on the HyperLogo bandwagon? Well, for one thing, HyperLogo is hard to find. A while back I started up HyperStudio 2.0. Where was Logo? I checked all the menu items. No luck. As a last resort I looked at the documentation. The *HyperLogo Users Guide* told me that the way to get into HyperLogo was to create a button and select "Use HyperLogo" from the Actions dialog box. I did this and found myself looking at a HyperLogo "script window." Following the Guide's suggestion, I typed **repeat 4 [forward 30 right 90]**, and closed the script window to return to HyperStudio. I clicked the button, and sure enough, a square

appeared on the screen.

I continued to follow the *User Guide* through more turtle graphics explorations. I was overwhelmed with nostalgia. HyperLogo is modeled on Apple Logo II, vintage 1983. OK. Been there. Done that. Now what?

There's more. HyperLogo extends turtle graphics into three dimensions, complete with blue/red glasses for viewing your 3D creations. Pretty neat, but what does this have to do with HyperStudio?

Finally, I came to a brief chapter about "Controlling HyperStudio from HyperLogo," which referred to samples in the HyperLogo Demo Stack. We move quickly from a description of how to use **readword** to a rather complicated pie graph program. The final sample in the Demo Stack was called "choices." It showed how a graphic image of a traffic light could be controlled via a Logo program, or instead, by using other features of HyperStudio. I preferred the Logo way, but I think that if I were a HyperStudio user who did not already know Logo, the sample would have convinced me that I didn't need to learn it.

Most people I spoke with while researching this article weren't familiar with HyperLogo. Those who knew about it generally found it hard to use. A frequent comment was that it was not well integrated into the overall HyperStudio environment. I'd have to agree. To use HyperLogo you enter a different world. To return to HyperStudio proper, you exit from HyperLogo. It's like walking back and forth from one room to another. The lack of integration extends to the documentation as well. The HyperStudio manual doesn't mention HyperLogo at all. The *HyperLogo Users Guide* is a separate book.

HyperLogo feels as if it has been attached to HyperStudio rather than built into it. The specific features that are needed to work with HyperStudio objects have been implemented as a collection of "Call Backs" that have been added on to a version of Logo that was designed for another environment in another era.

HyperStudio 3.0

My HyperStudio 3.0 upgrade arrived last week. The package includes a new book: *Exploring HyperLogo*, a tutorial written by Bill Lynn. It starts with navigation – how to get around a stack using HyperLogo. Next we find out how to hide, show, and move objects; manipulate text; and work with sound.

HyperStudio 3.0 includes a stack of samples for use with the *HyperLogo Tutorial*. A few new primitives have been added to HyperLogo, and there have also been some other minor changes. But the big improvement is in the documentation and samples. And it is a *big* improvement.

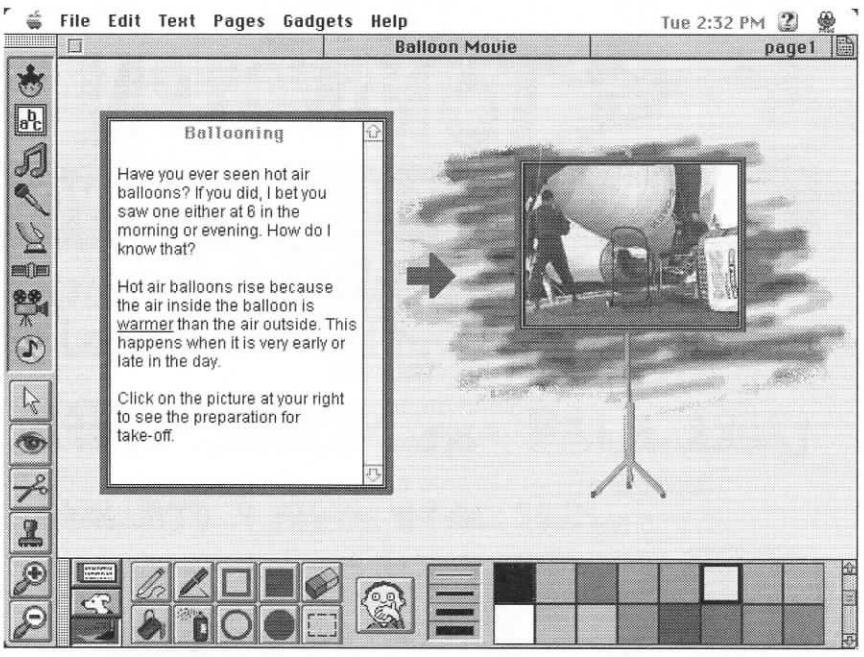
After going through the tutorial, I decided to poke around some of the other stacks that come with HyperStudio looking for more HyperLogo programs to play with. I didn't have to go far. On the first card of the Home Stack the buttons have Logo programs attached to them. But beyond that I was disappointed. HyperLogo was almost entirely absent from the rest of the samples. Since most of the material is contributed by students and teachers who use HyperStudio, this probably reflects what has been going on in the larger HyperStudio community.

I checked further on the Internet by joining the HyperStudio forum. I posted questions asking if anyone were using HyperLogo. Hardly anyone was. But HyperStudio 3.0 had only recently been released. The new documentation makes a big difference for me. Maybe it will for other people as well. Still, I'd be surprised to see a groundswell of enthusiasm for HyperLogo. There's something else going on here.

Two Different Worlds?

In the distant past, someone referred to Logo as having "no threshold and no ceiling." It is accessible to young children and to novices of all ages while still being useful for sophisticated explorations and projects. This oft-quoted phrase has guided the development of Logo-based learning environments. Until now, HyperLogo has had a very high threshold. With the recent release of

continues on the next page



MicroWorlds 2.0

The icons along the left edge of the screen are used to create and manipulate objects on the page. These include the text box, and the picture shown here, which is actually one frame of a QuickTime™ movie. MicroWorlds projects may also contain buttons, sliders, turtles, sounds, music, audio CD segments, and video disk sequences.

The lower part of the screen is alternately occupied by a Command Center for typing Logo instructions, a Shapes Center for creating and editing turtle shapes, and Drawing Center, which is shown here.

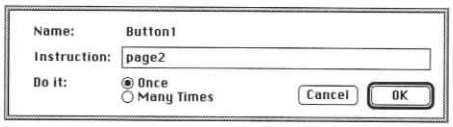
Logo and Hypermedia

continued from page 3

HyperStudio 3.0 the HyperLogo threshold has been lowered by having better documentation. Next, we can expect major improvements in the language itself. Still, Logo may never be in the HyperStudio mainstream because HyperStudio and Logo represent different approaches to software.

To illustrate this point, let's look at how buttons are created and modified in HyperStudio and in MicroWorlds. When I make a button in HyperStudio I first see a "Button Appearance" dialog box, which lets me choose from among scores of colors and dozens of shapes. Having worked through that, I see the "Actions" dialog box that appears on page 8 as part of Mike Westerfield's article. So many places to go! So many things to do! In fact, there are even more things to do since "New Button Actions..." is a doorway into a long and ever-growing list of possibilities.

Now here's what you see when you create a button in MicroWorlds:



The only choice is whether the instruction is run once or many times. Is this more limiting than HyperStudio? Hardly, since "Instruction" can be anything that MicroWorlds is capable of doing. In this case, the thing to do is to go to page2. Instead of presenting a list of ready-made choices, MicroWorlds invites you to write an instruction made up of Logo primitives and procedures that you or someone else wrote.

The Logo approach has not been without problems. Many versions of Logo start up with "Welcome to Logo" on the screen and then sit there, with cursor blinking, waiting to be told what to do. This is fine if you know what to say. But many a novice would just sit there for a while and then gingerly type "help." Logo would respond by saying "I don't know how to help," which is unfortunately true. At that point, the now annoyed novice would generally type something considerably less polite (which Logo wouldn't know how to do either).

Thankfully, many modern Logo environments are more helpful. MicroWorlds includes an on-line reference guide describing all the MicroWorlds Logo primitives. Still, the expectation is that help should come largely from sources outside the software.

The Logo way relies on a surrounding culture for ideas and support. Seymour Papert provided us with a model of this learning culture in *Mindstorms*.[1] He describes the Brazilian "Samba Schools" – not really schools, but social clubs in which people of all ages and levels of experience work together in both formal

continues on page 6

Logo and Hypermedia

continued from page 4

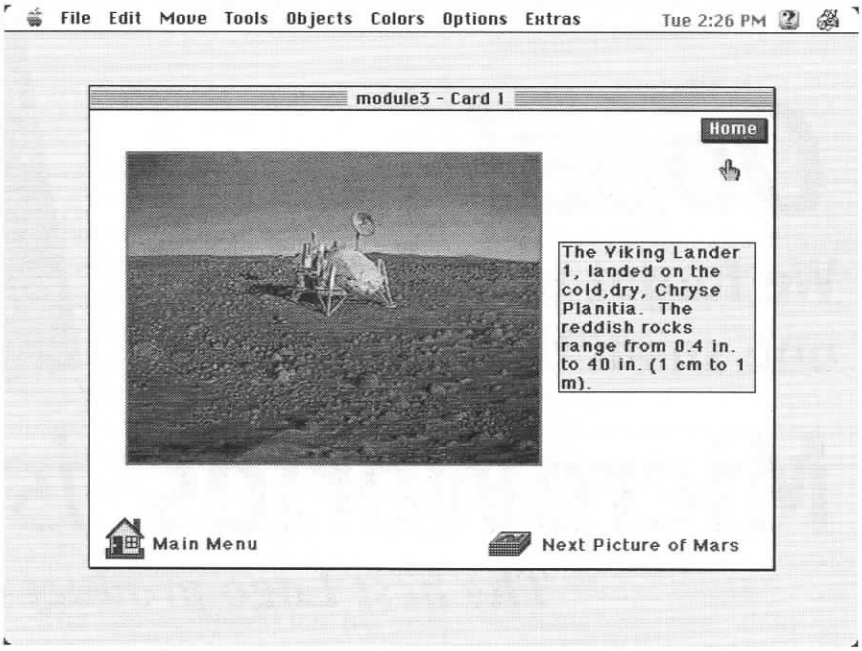
and informal ways to create elaborate pageants for Carnival. Unfortunately, Logo rarely finds itself in that sort of environment. Schools generally undermine the development of such a learning culture by excessively formalizing learning and segregating people by age and experience. It is no wonder that Logo is often on the fringes of educational technology, and that it appeals mostly to people who want to make fundamental changes in school.

HyperStudio finds itself in the same school culture as Logo. The HyperStudio solution is to build the helpfulness into the software. Even if there's no one around who knows about HyperStudio, the "what to do" is right in front of you, and a dog named Addy frequently pops up to help.

Roger Wagner, the creator of HyperStudio, has said to me about its design that "... it was our conscious intent to make it possible for anyone to accomplish a given end result as easily as possible," and that "HyperLogo is provided within that overall environment as one tool among many ... a student can choose to learn about HyperLogo when it provides an essential component to a larger overall project."

Yes, Logo should be used within a larger project. The goal is not to "do Logo" but to do something *with* Logo. But the design of HyperStudio necessarily creates a high threshold for HyperLogo. Since simple things are done without Logo, the first experience with Logo will be in the context of a hard problem. There is no opportunity to become familiar with Logo gradually, beginning with elementary activities.

In MicroWorlds people generally use Logo almost immediately. In a typical first project one might create a background with the drawing tools. Then a turtle could be turned into a dog or a car, aimed with the command **setheading 90** and set in motion by being told to go **forward 1** many times. Simple things are done with simple Logo. Complex things are possible with more advanced Logo.



HyperStudio 3.0

This HyperStudio project includes an imported picture along with explanatory text. Buttons are used to link to other cards in the project, and to launch various actions. These include playing sounds, activating audio and video disk sequences, playing QuickTime™ movies, and running Logo programs.

The menu bar is used to create, manipulate, and modify these objects, and to access drawing tools.

No threshold and no ceiling.

There are two issues here. One is a matter of personal preferences. For me, making choices from layers of menus is not easy. But I also recognize that many people prefer a different way. After all, HyperStudio *is* enormously popular and most commercial software applications, including the word processor that I'm using to write this article, are more like HyperStudio than MicroWorlds.

Second, there is the question of how a programming language fits into an application. If programming is reserved for the rare cases when you want to do something that can't be done with the existing features, what opportunity do you have to learn programming? Will you be able to program when you need to?

But even though HyperStudio and MicroWorlds represent divergent approaches to software design, in the larger scheme of things HyperStudio people and Logo people are in the same educational community. Both want to enable people to express their ideas, and to design and build personally meaningful projects. Both share a constructionist view of learning. It is no accident that Logo was chosen as the programming language for HyperStudio.

Roger Wagner says that with version 3.0 he has made a renewed commitment to Logo. The new tutorial and samples bear this out. Mike Westerfield is about to make major revisions to HyperLogo. They are both asking for input from the Logo community. I'm accumulating a long list of suggestions. I'm looking forward to the dialog. I encourage you to join in.▲

[1] Papert, Seymour, *Mindstorms*, Basic Books, New York, 1980, Chapter 8

Michael Tempel may be contacted at:
Logo Foundation
250 West 57th Street
New York, NY 10107-2228
tel: 212 765 4918
fax: 212 765 4789
michaelt@media.mit.edu

The 1996 Logo Summer Institutes

The Logo Summer Institutes are intensive five-day workshops that provide for an immersion in Logo theory and practice. The individualized approach of the Logo Summer Institute accommodates experienced Logo users as well as novices. The three components of the Summer Institute are:

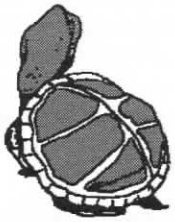
✔ **Logo lab**, where you have an opportunity to experiment and develop projects in one or more Logo-based computer learning environments. You learn skills and explore ideas while modeling activities that your students will do during the coming school year. Choose from among several strands including multimedia, turtle geometry, simulations, and game design.

✔ **Planning discussions**, organized by grade level and subject area, enable you to transfer your Summer Institute experience to the classroom.

✔ **Dialog groups** explore social, educational, and political issues raised by the technological transformation of school and the economy.

Logo St. Paul

Over the past 14 years the St. Paul Logo Project has provided a comprehensive professional development program for hundreds of elementary and secondary school teachers. A limited number of places are being set aside for people from outside the St. Paul Public Schools.



Summer at Spence

The Spence School, an independent school for girls located in the heart of New York City just steps from Manhattan's Museum Mile, is a leader in educational technology. Join the Spence staff for a week of Logo exploration and creation.

✔ Your registration includes
- all workshop materials;
- personal use of a computer – one machine per participant;
- use of a variety of Logo-based learning environments including MicroWorlds, Object Logo, StarLogo, Logo PLUS, UCBLogo, HyperStudio, PCLogo, and others;
- follow-up workshops in November 1996 and March 1997, each a day and a half long.

✔ Big discounts on purchases of Logo software, books, and materials.

✔ St. Paul registrants may also receive three graduate quarter credits from Hamline University.

Logo St. Paul
When: June 17 - 21 or August 19 - 23
Where: St. Paul, Minnesota
Tuition: $490 per person
$120 for Hamline University credit

If you are a teacher in the St. Paul Public Schools these registration procedures and fees do not apply to you. Instead, contact Ms. Geraldine Kozberg at 360 Colborne Street, 228-3631.

Summer at Spence
When: June 24 - 28
Where: New York City
Tuition: $790 per person

Attend **Logosium '96** on June 14th and stay for Logo St. Paul the following week! (See page 20 for Logosium information.)

Contact us for more information about Logo Summer Institutes and for registration materials. You may include your request on the order form on page 19. Also ask about organizing a Summer Institute at your school, and about other Logo workshop opportunities.

Logo Foundation
250 West 57th Street
New York, NY 10107-2228
telephone: 212 765 4918
fax: 212 765 4789
michaelt@media.mit.edu

HyperLogo:
Logo Like You've Never Seen It!
by Mike Westerfield

Multimedia has become a buzzword in the computer industry, so it's not surprising that many traditional implementations of Logo are starting to add multimedia features. What many Logo fans don't realize is that Logo is also the scripting language in one of today's most popular multimedia authoring tools.

In this article, we'll take a look at HyperStudio, a popular and powerful multimedia authoring program, and its scripting language, HyperLogo. You'll see a turtle fly and find out how to improve on $100 games with simple Logo programming.

Marrying Multimedia and Logo

There are two very different ways to approach tying multimedia and Logo together. To see what they are, let's back up and look at a similar problem that's been around for a while: word processing and text processing.

One approach to processing text is to start with a computer language and add powerful string manipulation features. Logo itself does this to some extent, but Snobol and Icon are the classic examples of this approach in text processing. The result is a powerful and flexible tool that can take text as input and do some marvelously complex things to the text. If you want to take an ASCII mailing list, reformat it, check for duplicates, change the capitalization, and so forth, this is the sort of tool you would pick. It's also possible to create business letters or format books – and that's been done.

Another approach is to start with a word processor and add a computer language that's generally not visible, but is available when needed. That's the approach used by Microsoft Word 6, for example. The result is a classic word processor that can handle very complicated text manipulation tasks, but that doesn't burden average users with a programming interface when all they want to do is write a letter.

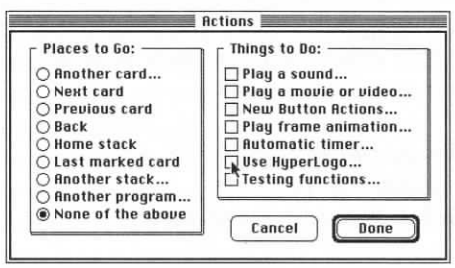
You see the same two approaches in the marriage of Logo and multimedia. Traditional Logos have added multimedia commands, giving a rich environment for displaying and manipulating multimedia objects. It's a great approach for teaching programming with interesting examples or manipulating multimedia, just as Snobol is a great language for manipulating text.

HyperStudio takes the other approach. It's a multimedia authoring tool, designed for creating computer documents with sound, pictures, text, hypertext links, movies, and even multimedia links to the Internet. And you can do all of this without scripting of any kind! HyperLogo is a scripting language, available for even more complex tasks. Frankly, though, most people who use HyperStudio never use HyperLogo. Those who do use HyperLogo use it most often for very simple tasks. In fact, that's one of the big differences between HyperStudio and HyperCard: HyperCard forces you into scripting right away, while most HyperStudio stacks are written with little or no scripting.

How HyperLogo Fits Into HyperStudio

So how does this work? Instead of scripting, HyperStudio relies on button actions – preprogrammed solutions to common multimedia tasks. You can attach these button actions to buttons, cards, or graphic objects. If you want to move to another card using a slow fade to black transition while playing a sound, you can do all of it by creating a button and clicking on a few choices in a dialog box. It's fast, painless, and you never see Logo.

In fact, Logo is just another choice



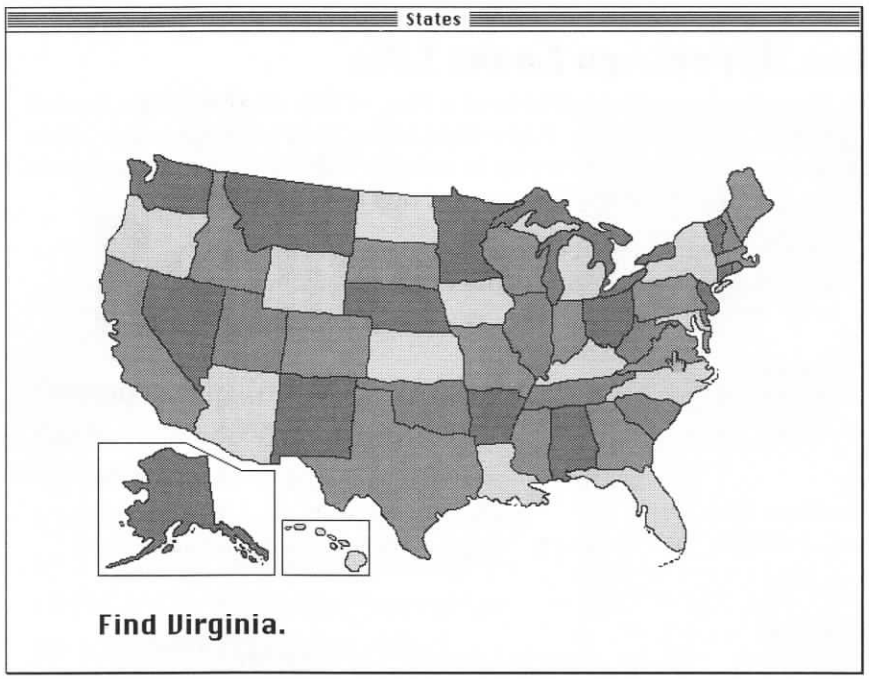
in a button actions dialog. If you select it, you get a text window where you can type a Logo script. This works just like a traditional Logo program. When the button is clicked, the Logo commands run just as if you typed them in a Logo text window. Procedures and global variables are entered in a common workspace that is available from any script, so scripts can share information. There's even a complete traditional style programming environment for prototyping and debugging Logo scripts without clicking on buttons to execute the program.

If you think about that for a moment, what you have so far is a Logo language that has very little to do with the multimedia program it's imbedded in. Up to now, about all you can really do with Logo is draw turtle graphic pictures on a card, **toot** a few notes, or write some text. HyperLogo's power as a multimedia tool really lies in a series of commands that are not in traditional Logo. These commands let Logo itself hide objects on the screen, read or write text to and from text objects on any card, search for a card with a specific text string, move to another card or stack, and so forth.

A typical example of marrying HyperLogo with multimedia is a state location quiz I wrote for my daughters in about two hours. I spent most of that time touching up the graphics – you can see why I never went for a career in art! In this stack, I pasted a map of the United States on a card, then created a series of 50 buttons, one for each state. HyperStudio was happy enough to create buttons using a bucket tool so they conform to the shape of the states, and to make the buttons invisible so the appearance of the button is controlled by the underlying painting. I attached a one-line script to each button. Here's the script for New Mexico:

```
Try "New\ Mexico
```

There's a fifty-first button on the card, too. It's a hidden button that is executed as soon as you load the stack. This button does all of the hard work. It starts by randomizing a list of state names, then displays one of them in a text field. You can see a state name in



the figure. The text field has the same color as the background, and no border, so all you see is the name floating on the background. When you click on a state, the **Try** procedure compares the name passed by the button to the correct state. If they match, the program makes a happy sound and moves on to the next state. If not, the program tells you which state you selected, and gives you another chance. After three wrong answers the correct state is flashed a few times.

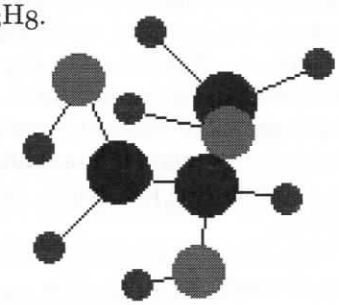
It's easy to see why HyperStudio is popular in schools. This kind of visual testing and game playing works with prereaders as well as kids who read well. It's also cost effective: This simple stack replaced a $100 dedicated game my daughter liked – and the stack works better than the original!

The 3D Turtle

For the most part, HyperLogo is a very standard implementation of Logo, in the same family as the old Apple Logo. I couldn't resist a twist or two, though! Late at night, while implementing turtle graphics, I sat playing with the commands. Rotate right. Rotate left. Rotate out...

Rotate out? Sure, why not. HyperLogo has a flying turtle! Instead of pinning the turtle to the flat computer screen, I added RotateOut, RotateIn, RollLeft amd RollRight. These commands give the turtle the ability to fly!

This is more than a gimmick. Consider the diagram of the molecule $C_3O_3H_8$.



Because it's drawn with a 3D turtle, I can redraw it quickly in a new orientation with just two simple commands:

```
RollLeft 45
Draw :C3O3H8
```

It's too bad this newsletter isn't in color on a CRT screen. The full color version of the molecule flipping around on the screen is rather striking. And because of the powerful combination of Logo and turtle graphics, it's also easy to create new molecules. I once demonstrated a stand-alone version of HyperLogo called 3D Logo to a group of Apple IIGS enthusiasts. I literally took a molecule from the audience (which I happened to know included a chemist), programmed the molecule, and created a movie of the molecule rotating around its axis during the talk. The whole process took about five minutes.

3D Logo has another trick that is, I admit, mostly gimmick–but what a gimmick! Instead of limiting you to a flat screen, HyperLogo also gives you the option of a true 3D display on any color Macintosh or Windows platform. Remember those old 3D movie glasses? Well, HyperLogo uses the same idea, drawing two versions of the image on a color screen. Put on a pair of the red-blue 3D movie glasses and the image literally jumps off the screen or dives into the monitor. I knew I had a great attention grabber when my daughter, then 7, reached out to run her fingers through the objects floating in space!

Without color, I'm not even going to try to show you what it looks like. There is a 3D sample with HyperStudio, though. Check it out sometime.

Why Logo?

Now that you know a little about HyperLogo and how it fits into HyperStudio, I can go back and deal with a question that logically should have been answered right away. Why Logo? After all, no other multimedia tool uses a standard language as its scripting language. Why not invent yet another new semi-English programming language to confuse people with?

Well, I sort of answered the question with that comment. As a language implementor, I always look for an existing language that will do the job before trying to create a new one. An existing language with a proven track record is less likely to have subtle traps that prevent it from working effectively. With a common language, you also get the benefit of a group of people who already know how to use the language.

When Roger Wagner, creator of HyperStudio, asked me for a recommendation for a scripting language, he might have expected me to recommend C, Pascal, or the like. After all, I've written compilers for both of those. A scripting language for a multimedia program cries out for list processing, text manipulation, and artificial intelligence, though. The first language I thought of was LISP, which has also been used as a script-

continues on the next page

HyperLogo

continued from page 9

ing language. But HyperStudio started on the Apple IIGS and was most popular at the time in K-8 education. That's the same territory where Logo is popular. The choice was so obvious that I found out later another programmer made exactly the same recommendation!

There was one point I admit I missed. I assumed when HyperStudio and Logo were joined, Logo programmers would jump on the new tool and begin creating fabulous stacks that married multimedia and artificial intelligence. I expected to be bombarded with suggestions and demands from Logo programmers who wanted to push the language faster than I could develop it. I expected that by now, there would be a regular column here dealing with HyperLogo.

So, to paraphrase an old 60's song, "Where have all the Logo programmers gone?" Roger Wagner speculates that HyperLogo is now in the hands of more people than any other implementation of Logo ever. Looking at his sales of HyperStudio, I suspect he's right. As HyperStudio migrates to Windows, I'm sure that if he isn't right already, he soon will be. HyperLogo gives you folks, the Logo community as a whole, a chance to promote this great language through an entirely new medium. I'd like to hear from you. More than that, I'd like to see you help me promote the use of HyperLogo for more than just hiding more than one graphic object at a time!

The Future of HyperLogo

HyperLogo is not a finished language. New developments have been slow for the past year simply because most of my energy has been devoted to porting the language to Windows, but with that port complete, HyperLogo is about to undergo a whole new set of changes.

The weakest link in HyperLogo today is the user interface. That's the next major thrust. If you're interested in helping develop a whole new way for non-programmers to learn and use Logo, join me on America Online for the development and beta test of our new user interface.

I'm already planning a series of changes that will take Logo beyond the boundaries of HyperStudio. HyperStudio itself can already control Laser Disc players and CD ROM players, but controlling them with more options from HyperLogo is coming. I'm also working with other companies to bring control of other real-world devices to Logo. I'd love to talk to you if you're interested in robotics and scientific test instruments such as barometers and force sensors.

The bottom line is that comments, and yes, complaints, from customers have been the driving force in adding new features to HyperLogo so far, and I hope they are in the future. Consider this an open invitation to help shape the future of the most commonly available Logo anywhere.▲

Mike Westerfield may be contacted at:
Byte Works, Inc.
8000 Wagon Mound Dr. NW
Albuquerque, NM 87120
505 898 8183
MikeW50@aol.com

What HyperLogo Looks Like

In this example, the viewer sees a map of the northwestern United States with Idaho colored in. A text field tells a little about the state, then asks the viewer to enter the name in another text field. When the viewer clicks on a button to check the answer, this script runs.

```
to CheckAnswer :correct :student
If :correct = :student [Correct] [NotCorrect]
end

to Correct
SetFieldText [] "Result 'That\'s right! Now for your
next lesson...'
end

to NotCorrect
SetFieldText [] "Result 'Oops. Maybe you should read
the hint.'
end

CheckAnswer "Idaho GetFieldText [] "answer
```

After entering some procedures, the script arrives at an executable line. **GetFieldText**, in the last line of the script, is a function that reads the text from a HyperStudio text field – in this case, a text field on the current card named "answer. (The empty list is a place-holder. You can use a name instead, and grab text from another card.)

The result is passed to **CheckAnswer**, along with the correct answer. It checks the answer, and in this case, just prints a message. You could just as easily use a command like**MoveToCard "Review** for an incorrect answer, popping over to a multimedia remedial training card, or **MoveNext** to move to the next card, continuing the main line of the multimedia lesson.

Printing the message for a correct or incorrect message is handled with **GetFieldText**'s opposite number, **SetFieldText,** which can dump any Logo value, including a list, into a HyperStudio text field.

3D Logo, a stand-alone cousin of HyperLogo for the Apple IIGS is available from Byte Works at the above address.

HyperStudio is available for Macintosh and Windows from:
Roger Wagner Publishing, Inc.
1050 Pioneer Way, Suite P
El Cajon, CA 92020
619 442 0522
RWagnerInc@aol.com

Meet Roamer!

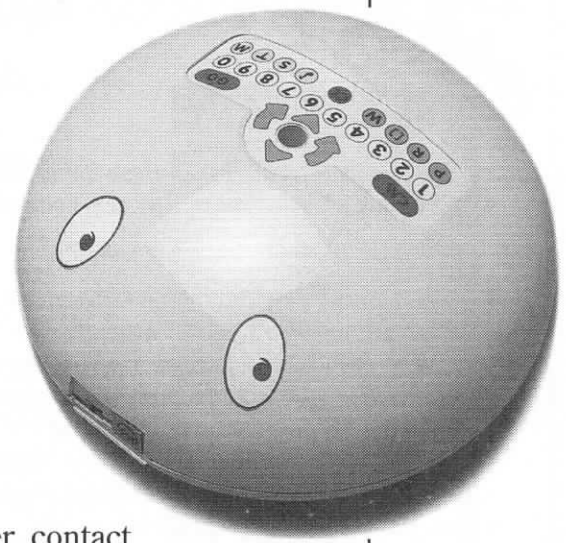
Introducing the very latest in Logo technology - Roamer! Combine the magic of the world of robotics with the fun of Logo for even the youngest child with Roamer, an easy-to-use and friendly robot.

Begin Logo explorations with a true hands-on experience.

Roamer is the ideal way to introduce Logo commands in your classroom. It brings Logo to life in a friendly and tangible way. Since Roamer has Logo already on-board, it's like having a computer dedicated to your Logo lessons. Children can touch, feel, and follow the Logo turtle as it moves about.

Roamer is designed to be sturdy, with few moving parts and a simple and friendly shape. It's easy to use - with a brightly colored touchpad featuring single key-stroke commands. Roamer is lightweight and robust. Roamer is battery-powered and offers you hours of Logo adventures - exploring, discovering and building.

Available kits make it possible to customize Roamer - eyes, nose, ears, a tail - let the children design their own! There are even four different shells to change Roamer's color to red, yellow, white, or green. Insert a colored marker pen and watch Roamer draw fascinating designs.



Roamer can also play music. Pitch, duration, and tempo add an extra dimension to any Logo lesson. A simple and clear User Guide and Activity Book accompany Roamer and will start your students on their way to hours of Logo fun and learning. **$299.95**.

For more information or to order your Roamer, contact Harvard Associates at 1-800-774-LOGO or fax 1-800-776-4610.

HARVARD
ASSOCIATES, INC.

10 Holworthy Street • Cambridge, MA 02138 U.S.A. • Phone (617) 492-0660
Fax (617) 492-4610 • Compuserve 70312,243 • Internet pclogo@harvassoc.com

Book Briefs

In this issue of *Logo Update* we briefly review three books which offer ideas and material that can be used for classroom Logo projects.

101 Ideas for Logo, by Dorothy Fitch (Terrapin Software, Cambridge, MA, 1993) provides a wealth of ideas and Logo project starters. Fitch is a teacher and writer who has a very good sense of what can work in real classroom situations. To borrow a phrase from Seymour Papert, here is Logo in "mind sized bites." Ideas are each presented in a single page, ready to be handed out in class. These are self-contained short activities, but may become starting points for larger projects. We are told what Logo skills are needed for each activity. This is helpful, although the more general labeling of projects as beginning, intermediate, or advanced is unnecessary tracking.

101 Ideas for Logo is heavy on "Classic Logo" activities, mostly involving turtle graphics, and is especially well matched to Terrapin Logo and PC Logo. The projects may be adapted for use with other versions of Logo. You may find this book to be a good supplement to the materials that came with your software.

Teaching With Logo, by Molly Watt and Daniel Watt (Addison-Wesley, Menlo Park, CA, 1986) is a story about Lynn, a sixth grader, four of her classmates, and their teachers. We follow Lynn's Logo work through eight class sessions. We share her explorations, ideas, questions, and bugs. We have a full record of her Logo programs and designs along with her journal entries and teachers' notes.

Except for the teachers, who are the Watts, the characters are actually composites drawn from the many students the authors have worked with over the years. This idealization provides a clean and streamlined way of presenting us with a model Logo learning environment, but one that is entirely realistic and achievable.

Teaching With Logo focuses on turtle graphics projects, but the teaching strategies, and especially the ideas about building a Logo learning culture, are applicable to any domain of Logo work, and to teaching and learning in general.

Minds in Play: Computer Game Design as a Context for Children's Learning, by Yasmin Kafai (Lawrence Erlbaum Associates, Mahwah, NJ, 1995) is based on the author's doctoral thesis at MIT. She provides a theoretical framework for looking at learning through design,

Order **Logo PLUS** from the Logo Foundation.
Turn to page 18.

and a detailed account of children's work in creating computer games. Kafai worked with a class of fourth graders at the Hennigan School in Boston. They used LogoWriter to create computer games aimed at teaching younger children about fractions. *Minds in Play* was not written as a source of classroom activities, but the detailed descriptions of children's Logo work provide ample information for teachers who want to initiate the same kinds of projects with their own students. And unlike most activity books, this one also includes the research you may need to "justify" what you're doing. ▲

101 Ideas for Logo, Teaching with Logo, and *Minds in Play* may be obtained from the Logo Foundation. Turn to page 18 for pricing and ordering information.

Order **PC *logo*** from the Logo Foundation. Turn to page 18.

Public Domain Logo

There are some versions of Logo that are in in the Public Domain, which means that they may be freely copied and distributed.

UCBLogo, written by Brian Harvey, lacks the fancy graphics of modern commercial Logos and it doesn't make much use of the Macintosh / Windows type of user interface. But it is a complete implementation of the Logo language and includes advanced features not found in any other version. It is ideal for use with Harvey's book *Computer Science Logo Style*. UCBLogo 3.3 is the most recent version for both Macintosh and MSDOS computers.

MSWLogo, written by George Mills, is based on UCBLogo and also takes advantage of features of the Windows environment in which it lives. Version 4.2 has just been released.

Both versions are distributed along with their user manuals and source code.

UCBLogo and MSWLogo are available at these Internet sites.

UBCLogo:	`ftp://anarres.cs.berkeley.edu/pub/ucblogo/`
MSWLogo:	`http://www.ultranet.com/~mills/`
Both:	`ftp://cher.media.mit.edu/pub/logo/`

The Turtle Discovery Book, by James Muller, newly revised for use with MSWLogo is also at the MSWLogo site.

If downloading is not convenient, you may obtain UCBLogo and MSWLogo on diskette from the Logo Foundation for a nominal service charge plus shipping and handling. Turn to page 18 for ordering information.

Continua...

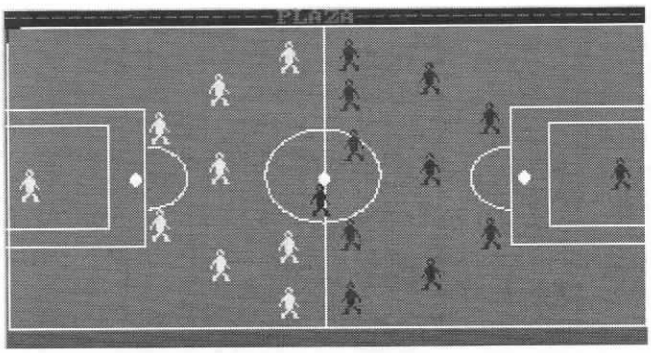
First reported by Gabriela Ramirez in the winter 1995 issue of *Logo Update*, The **Educational Telecommunications Network** in Costa Rica is now in full swing. The *Revista Juvenil*, an electronic magazine, published its first edition in July 1995 with the contributions of 450 elementary school students from 15 schools located in all the provinces of Costa Rica.

The magazine contains news and views about politics, society, science, sports, and literature; and children's poems, stories, and LogoWriter projects.

A second edition was published in October 1995. You may see the *Revista Juvenil* at the World Wide Web site of the Omar Dengo Foundation:

http://www.fod.ac.cr/

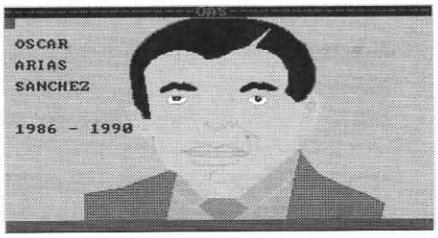
Order **CRYSTAL RAIN FOREST** from the Logo Foundation. Turn to page 18.



This illustration, which was created using LogoWriter, accompanied an article about soccer which appeared in the sports section of the October 1995 edition of the *Revista Juvenil*.

PROYECTO PRESIDENTES DE COSTA RICA

OSCAR ARIAS SANCHEZ



Nació en Heredia el 13 de setiembre de 1941. Ingresó al partido Liberación Nacional cuando apenas tenía 20 años de edad y de allí en adelante escaló posiciones políticas rápida y decisivamente.

Obtuvo el internacional e importante Premio Nobel de la Paz en 1987 por su iniciativa desplegada para obtener la paz.

Construyó el hermoso edificio piramidal de la Contraloría frenta a la Sabana, el Parque de la Paz en el sur de la capital, y otras importantes obras públicas. Se aprobaron la Ley de Sicotrópicos contra las drogas, la Ley de Promoción de la igualdad de la Mujer, el Banco Hipotecario de la Vivienda (BANHVI), la Sala Cuarta, y el Código Ambiental.

This illustration and text are from a LogoWriter project about the presidents of Costa Rica. It is on the World Wide Web site of the Omar Dengo Foundation, where you will also find information about other aspects of the Programa de Informática Educativa.

Are you still using that old Apple II?

Make the best of it with these special purpose toolkits that work with your existing Logo to provide numerous activities and project ideas. Each package includes software and extensive written materials.

For use with LogoWriter™

- **LogoWriter Hypermedia Tools**
  Create hypermedia documents which include graphics, text, and sound. Control a laserdisc player (optional).
- **LogoWriter Language Arts Tools**†
  Contains activities to develop writing skills and encourage self-expression.
- **LogoWriter Graph Tools**†
  Create bar graphs, pie charts, line graphs, and pictograms.
- **LogoWriter for Special Needs**
  Includes single key interface, large type word-processing, and other supports for students who are physically challenged or have learning disabilities.

For use with Terrapin™ Logo or Logo Plus™

- **Logo Data Toolkit**
  Take surveys, create data bases, generate graphs and charts.

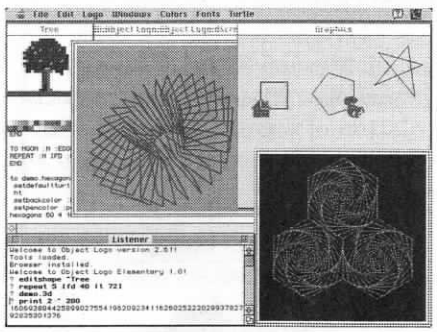
only $9.95 each

†also available for use with Apple IIGS or MSDOS LogoWriter

Turn to page 18 for ordering information.

Order *Object* **Logo** from the Logo Foundation. Turn to page 18.

*Object***Logo**™ for the Macintosh® - The Logo Language for All Ages!



Regarded by educators as the most powerful Logo on the market, *ObjectLogo* is now also one of the easiest Logo languages to learn and use. Thanks to the 180-page highly acclaimed tutorial, *Logo for the Macintosh,* by Harold & Amanda Abelson. Whether your interest is for home or school, give *ObjectLogo* a try. The Student Edition (includes the tutorial) is well-suited for the beginner. The Full Version (includes the tutorial and the Reference Manual) is for a more serious exploration of programming on the Macintosh. Lab Packs include both the tutorial and Reference Manual.

Features Include:

- Support for 32-bit Addressing
- Extensive Math Features
- Advanced List Processing
- Full Macintosh Color
- Complete Object System
- Fast Incremental Compiler
- Supports Multiple Windows
- LEGO® Logo Support
- Comprehensive Tutorial

Student Edition	Full Version	Five Lab Pack	Ten Lab Pack	Twenty Lab Pack
$75.00	$195.00	$341.00	$487.00	$585.00

Paradigm SOFTWARE

P.O. Box 670 • Cambridge, MA 02142
tel: (617)576 7675 • fax: (617) 576 7680 • info@paradigmsoft.com

StarLogo1.5 is Here

In the Winter 1995 issue of *Logo Update*, Carol Sperry reviewed Mitchel Resnick's *Turtles, Termites, and Traffic Jams*, which describes explorations using a massively parallel version of Logo known as StarLogo.

A Macintosh version of StarLogo was developed by Mitchel Resnick, Brian Silverman, Andy Begel, and Randy Sargent at MIT and was made available for educational and research purposes. A new improved version 1.5 has just been released. Changes include:

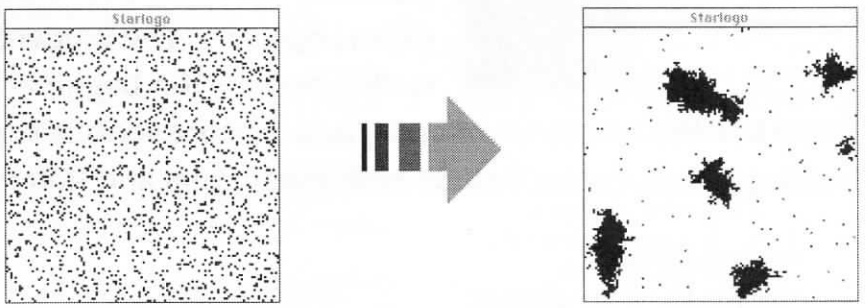
- StarLogo now has Power PC native code!
- Programs can now control "plot pens" for plotting data in the Plot Window.
- You can now modify underlying StarLogo parameters (such as the number of patches, the size of patches, and the maximum number of turtles).
- StarLogo can now load pictures from PICT files and grab images from QuickTime-compatible cameras.
- StarLogo now has mouse primitives, so that you can interact with StarLogo programs using the mouse.

...and much much more!

To run StarLogo version 1.5, you need a Macintosh with at least a 68030 processor and at least 5 Mbytes of free RAM.

StarLogo 1.5 (along with supporting documentation and new sample projects) is available via ftp from chelona.media.mit.edu (user: sldist, password: MrSabb). Download the file starlogo-1.5.sea.hqx from /pub/starlogo. You need to unBinHex the application using a utility such as Compact Pro or Stuffit – or download with Fetch, which will automatically unBinHex the application. You will get a self-extracting .sea file. When you double-click this file, it will create a new folder called StarLogo 1.5 with the application, documentation, and sample projects inside. If you are not able to do this, or if this entire paragraph makes no sense to you, look at page 18 of this newsletter to find out how to obtain a copy on disk.

Turtles, Termites, and Traffic Jams, by Mitchel Resnick, is a great book to have if you're working with StarLogo. Many of the projects described in the book, such as the simulation of termite behavior illustrated below, are included as samples with the software. Turn to page 18 of this newsletter to find out how to order a copy.



Hundreds of wood chips are scattered at random. A hundred termites wander around randomly, each following a set of simple rules: "If you're not carrying a chip pick up the next one you encounter. If you are carrying a chip, put it down near the next chip you run into. Then scoot away and repeat the process." After a while the chips are mostly in a few large piles although the rules the termites follow say nothing about building piles.

Order **Mach Turtles *Logo*** from the Logo Foundation. Turn to page 18.

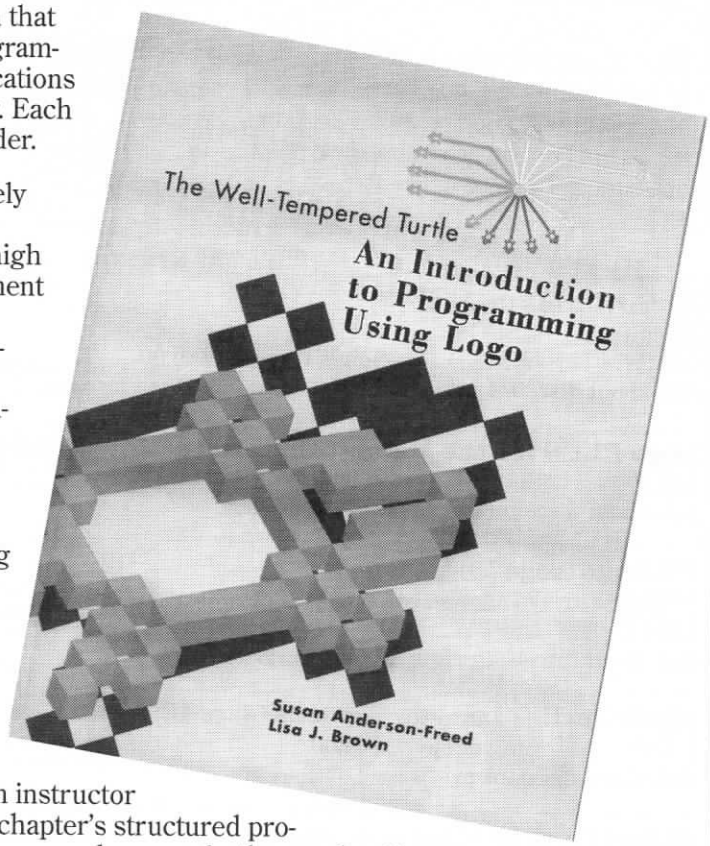
Order **The Well-Tempered Turtle** from the Logo Foundation. Turn to page 18.

The Well-Tempered Turtle

An Introduction to Programming Using Logo

by Susan Anderson-Freed and Lisa J. Brown

The Well-Tempered Turtle is a new curriculum that uses Logo as a means of testing and exploring programming concepts. It emphasizes learning Logo applications and highlights Logo's unique programming power. Each chapter is independent and may be used in any order.



The Well-Tempered Turtle has been extensively field-tested in introductory college level computer science courses and is appropriate for students of high school and college age. By utilizing Logo to implement examples, *The Well-Tempered Turtle* has students quickly writing their own programs to explore computer science concepts. Students build on simple introductory programs to explore increasingly complex subjects, progressing for example from line drawings to fractals and bit-mapped graphics.

The Well-Tempered Turtle also provides a complete introduction to computer science covering such topics as data types, control structures, graphics, natural language processing, and music. Appendices provide supplementary information on the history of computers, mathematics and grammar.

Since *The Well-Tempered Turtle* contains more material than can be covered in a semester, an instructor can pick and choose the topics to emphasize. Each chapter's structured progression encourages students to learn at their own pace and pursue further exploration.

The Well-Tempered Turtle is written by Dr. Susan Anderson-Freed and Dr. Lisa J. Brown, Professors of Computer Science at Illinois Wesleyan University. Together they have more than 27 years' experience teaching mathematics, programming and computer science. Their Logo courses are both highly demanding and in high demand among students at Illinois Wesleyan, and always fill immediately.

250 pages. **$49.95**

To order *The Well-Tempered Turtle,* please call 1-800-774-LOGO
or fax 1-800-776-4610

HARVARD
ASSOCIATES, INC.

10 Holworthy Street • Cambridge, MA 02138 U.S.A. • Phone (617) 492-0660 • Fax (617) 492-4610
Compuserve 70312,243 • Internet pclogo@harvassoc.com

Software and Books

Many of the items listed here are described elsewhere in this issue of Logo *Update*. Turn to the pages indicated for more information about these products.

New Lower Prices

(Guaranteed through May 31, 1996)

The Logo Foundation now offers all commercial Logo software at below retail rates. The prices shown here reflect these discounts. Discounts are also available on lab packs and site licenses. Please contact us for current prices.

Even larger discounts apply when software is purchased by participants in Logo Foundation workshops and Summer Institutes, such as those described on page 7, and in conjunction with workshops we conduct in your school or district. Contact us for details.

MicroWorlds (page 5)

Macintosh	MSDOS		
LSMWM	LSMWD	Single Copy	$89.00

Ask about discount prices for Lab Packs and Site Licenses.

Logo PLUS for the Macintosh (page 12)

LSLP	SingleCopy	$89.00

Ask about discount prices for Site Licenses.

PC Logo (page 13)

MSDOS	Windows		
LSPCD	LSPCW	SingleCopy	$89.00

Ask about discount prices for Site Licenses.

Mach Turtles Logo for Windows (page 16)

LSMT	SingleCopy	$45.00

Ask about discount prices for Lab Packs.

Object Logo for Macintosh (page 15)

LSOLSE	Student Edition	$ 69.00
LSOLFV	Full Version	$179.00

Ask about discount prices for Lab Packs.

Crystal Rain Forest (page 14)

Macintosh	MSDOS		
LSCRM	LSCRD	Single Copy	$45.00
LSCRMS	LSCRDS	School Edition	$55.00

Ask about discount prices for Site Licenses.

UCBLogo (page 13)

Macintosh	MSDOS	
LSUCBM	LSUCBD	$2.00

MSWLogo for Windows (page 13)

LSMSW	$2.00

StarLogo for Macontosh (page 16)

LSSL	$2.00

101 Ideas for Logo

by Dorothy Fitch

Lots of great Logo ideas for less than 20¢ apiece! See Book Briefs on page 12.

LB113SC	single copy	$19.95
LB113SL	site license	$50.00

Teaching With Logo

by Molly and Daniel Watt

This is a unique source book offering educators and parents a wealth of information about using Logo. Look at Book Briefs on page 12.

LB111	$15.00

Minds in Play: Computer Game Design as a Context for Children's Learning

by Yasmin Kafai

"A mine of ideas for teachers in search of computer projects for their students or themselves" - Seymour Papert. Read Book Briefs on page 12.

LB110P	$24.95 (paperback)
LB110H	$59.95 (hard back)

The Well-Tempered Turtle

by Susan Anderson-Freed and Lisa J. Brown

Turn to page 17 for a detailed description of this comprehensive curriculum guide for high school and college students.

LB112	$49.95

Computer Science Logo Style

by Brian Harvey

The best tutorial available for learning Logo. A good companion to UCBLogo and MSWLogo. See page 13.

Volume 1: Intermediate Programming

LB114	$22.95

Volume 2: Projects, Styles, and Techniques

LB115	$21.95

Turtles, Termites, and Traffic Jams

by Mitchel Resnick

The book about StarLogo. Look at page 16.

LB116	$24.95

Roamer

Look at the description of this creature on page 11.

LROAM	$279.00

New low price!

Logo Toolkits (page 15)

LTKHM	LogoWriter HyperMedia Tools	$9.95
LTKLA	LogoWriter Language Arts Tools	$9.95
LTKG	LogoWriter Graph Tools	$9.95
LTKSN	LogoWriter for Special Needs	$9.95
LTKLD	Logo Data Toolkit	$9.95

Order Form

To place an order:

1. For each item you want, fill in the form below with:
   - the item code (it begins with L)
   - name or description of the item
   - the quantity you want
   - the price for the quantity you want
2. Add up the cost of all your stuff and enter it under sub-total.
3. Figure out the shipping and handling charge. Regular shipments are via UPS or US Postal Service.
   United States: 5% of the sub-total with a minimum charge of $5.00
   Canada and Mexico: 10% of the sub-total with a minimum charge of $10.00
   The rest of the World: Inquire before ordering, stating whether you want us to ship by air or by sea.
   Rush Orders: We can ship overnight or second day delivery. Inquire about costs before ordering.
4. Add the shipping and handling charge to the sub-total and enter that amount under TOTAL.
5. Please consider making a tax-deductible contribution to the Logo Foundation. Thanks.
6. In the "Send me" section below, check off the items that you want more information about.
7. Send this form with a check or school puchase order to:

Logo Foundation
250 West 57th Street
New York NY 10107-2228
telephone: 212 765 4918
fax: 212 765 4789

Payment must be in US dollars.
Purchase orders and inquiries may be faxed.

Send me:

Information and registration materials for the Logo Summer Institutes:
❑ Logo St. Paul ❑ Summer at Spence

❑ Information about Logo Foundation Professional Development Services

❑ A free subscription to *Logo Update*

Place my order for:

item code	description	quantity	price
		sub-total	
		shipping & handling	
		tax-deductible contribution to the Logo Foundation	
		TOTAL	

Name _____

Address _____

_____

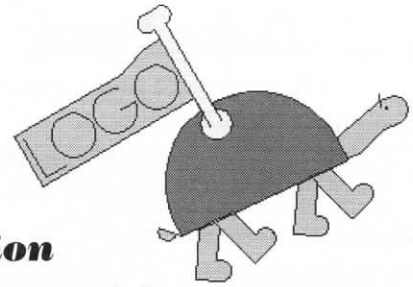
City _____ State _____ Zip _____

Telephone _____ Fax _____ e-mail _____

Send this form to:
Logo Foundation
250 West 57th Street
New York NY 10107-2228

Logosium '96

St. Anthony Park Elementary School
St. Paul, Minnesota
Friday, June 14, 1996



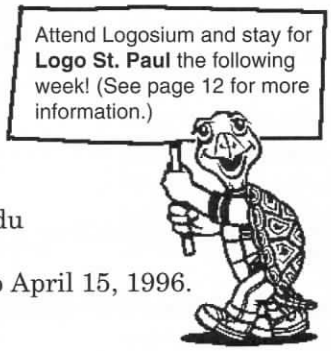
Call for Participation

The third annual Logosium will be a day of Logo workshops, discussions, and presentations hosted by the world-renowned St. Paul Logo Project. This year Logosium will include sessions conducted by students sharing their projects with other students and with adults.

Sessions may be one-hour presentations or panel discussions, or two-hour hands-on workshops, on any topic of interest to the Logo community.

If you wish to offer a session, submit a one-page description of what you have in mind to:

Marian Rosen & Michael Tempel
c/o Logo Foundation
250 West 57th Street, Suite 2228
New York, NY 10107-2228
Telephone: 212 765 4918 Fax: 212 765 4789
e-mail: mbrosen@oui.com michaelt@media.mit.edu



The deadline for submissions has been extended to April 15, 1996.

For registration and hotel information contact:
NECC '96
1787 Agate Street
Eugene, OR 97403-1923
Telephone: 503 346 2834 Fax: 503 346 5890
e-mail: necc@oregon.uoregon.edu

Logosium is an NECC '96 post-conference activity sponsored by the Logo Foundation and ISTE's SIG Logo.

LOGO USERS GROUPS

Long Island Logo Users Group
Contact: Marilyn Tahl
516 333-4018 (evenings)
516 627-8110 (days)

Los Angeles Logo Users Group
Contact: Carolina Goodman
Country School
5243 Laurel Canyon Blvd.
North Hollywood, CA 91607
818 769-2473

Logo Anonymous
Contact: Marian Rosen
Conway School
9900 Conway Road
St. Louis MO 63124
314 993-2878

New York Logo Users Group
Contact: The Logo Foundation
212 765-4918

Philadelphia Logo Users Group
Contact: Mel Levin
Prince Hall School
Godfrey and Gratz Avenues
Philadelphia PA 19141
215 276-5369

Shoreline Logo Users Guild
Contact: Jeff Sandys
1306 NW 201st Street
Shoreline, WA 98117-2146
206 294 6450

Logo Foundation
250 West 57th Street • New York, NY 10107-2228

Nonprofit Org.
U.S. Postage
PAID
New York, NY
Permit #6378