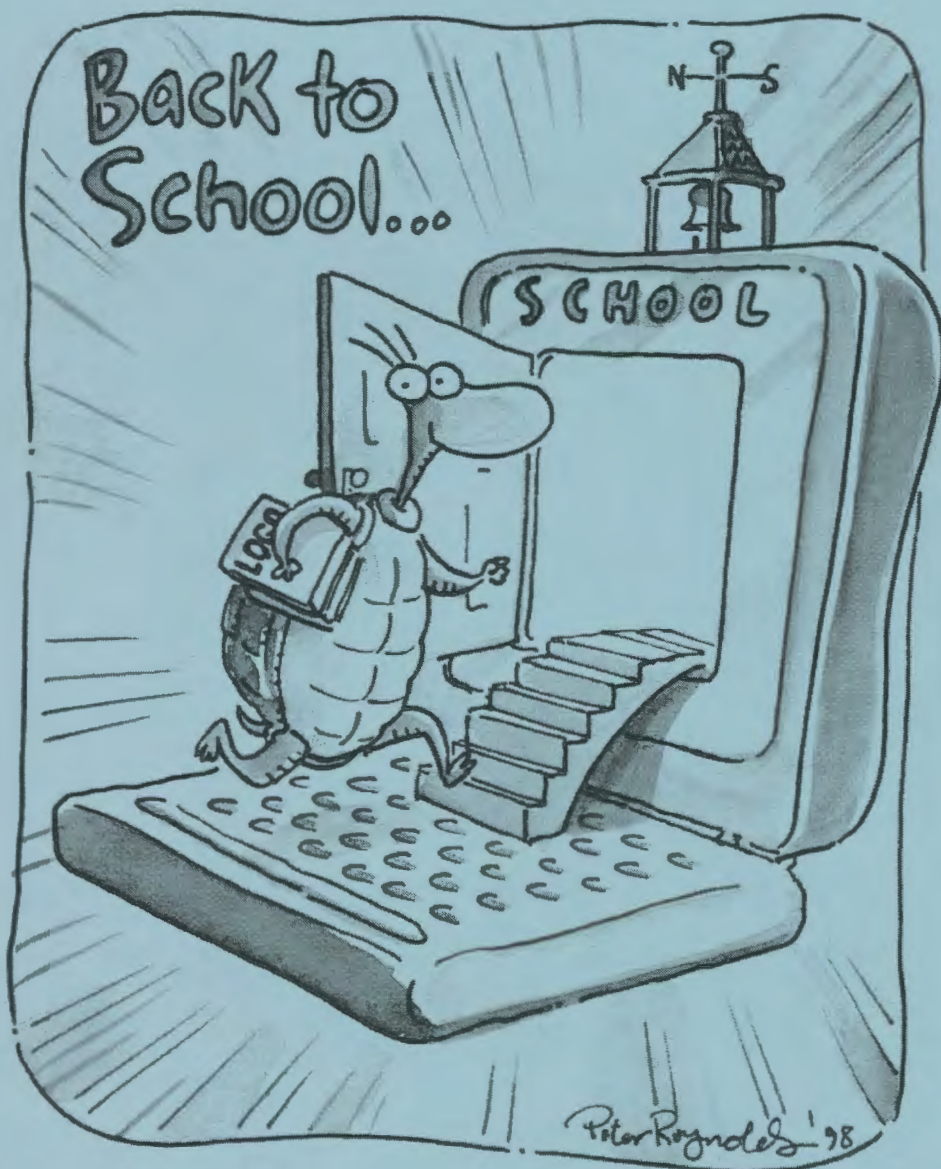


Logo Exchange

Journal of the ISTE Special Interest Group for Logo-Using Educators



BACK TO SCHOOL!

INSIDE

- What I Did on My Three Summer Vacations
- A Mayan Pyramid
- Build Your Own Virtual Pet With MicroWorlds
- Drawing Polyhedron Patterns
- More from the President's Panel
- Surfing the Net—Rehabilitating a Metaphor
- New Year's Resolutions
- Book Review, Logo Online, Teacher Feature

Logo Exchange

Volume 17 / Number 1

Editorial Publisher

International Society for Technology in Education

Editor-in-Chief

Gary S. Stager, Pepperdine University
logoexchange@moon.pepperdine.edu

Copy Editing, Design, & Production

Ron Richmond

Founding Editor

Tom Lough, Murray State University

Design, Illustrations & Art Direction

Peter Reynolds, Fablevision Animation Studios
pete@fablevision.com

Contributing Editors

Dr. Douglas Clements, SUNY Buffalo
Dr. Carolyn Dowling, Australian Catholic University
Alan Epstein, Metasoft
Dr. Brian Harvey, U.C. Berkeley
Daniel E. Kinnaman, University Affiliates
Dr. Julie Sarama, Wayne State University

International Editor

Jeff Richardson, Monash University, Australia

International Editor Emeritus

Dennis Harper, Olympia, Washington School District

SIGLogo Officers

Stephen Sesko, President
Jeff Richardson, Vice-President
Hope Chafian, Secretary/Treasurer
Gary S. Stager, Editor

Director of Advertising Services

Judy Stickney

1998-1999 ISTE BOARD OF DIRECTORS

ISTE Executive Board Members

Lynne Schrum, President *University of Georgia-Athens (GA)*
Heidi Rogers, President-Elect *University of Idaho*
Cheryl Lemke, Secretary *Milken Family Foundation (CA)*
Michael Turzanski, Treasurer *Cisco Systems, Inc. (MA)*
Chip Kimball, At Large *Lake Washington School District (WA)*
Cathy Gunn, At Large *Northern Arizona University*

ISTE Board Members

Larry Anderson *Mississippi State University*
Jose Calderoni *ILCE, Mexico*
Penny Ellsworth *Western Springs School District 101 (IL)*
Marianne Handler *National-Louis (IL) University*
Dennis Harper *Olympia School District (WA)*
Jorge Ortega *FACE/Leon County SD (FL)*
Neal Strudler *University of Nevada, Las Vegas*
Sue Waalkes *Upper Dublin School District (PA)*
Peter Wholihan *Sts. Paul & Peter School, Virgin Islands*

ISTE Committees

Lajeane Thomas *Accreditation and Standards*
Dave Brittain *Awards*
Cathy Gunn *Distance Learning*
Michael Turzanski *Finance*
Paul Resta and Gerald Knezek *International*
Jenelle Leonard *Minority Affairs*
Lary Smith *Policies and Procedures*
Gwen Solomon *Publications*

ISTE Executive Officer for Research and Development

David Moursund

Logo Exchange is published quarterly by the International Society for Technology in Education Special Interest Group for Logo-Using Educators. *Logo Exchange* solicits articles on all aspects of Logo use in education.

Submission of Manuscripts

Manuscripts should be sent by surface mail on a 3.5-inch disk (where possible). Preferred format is Microsoft Word for the Macintosh. ASCII files in either Macintosh or DOS format are also welcome. Submissions may also be made by electronic mail. Where possible, graphics should be submitted electronically. Please include electronic copy, either on disk (preferred) or by electronic mail, with paper submissions. Paper submissions may be submitted for review if electronic copies are supplied on acceptance.

Send surface mail to:

Gary S. Stager
21825 Barbara St.
Torrance, CA 90503 USA

Send electronic mail to:

logoexchange@moon.pepperdine.edu

Deadlines

To be considered for publication, manuscripts must be received by the dates indicated below.

Vol. 17, No. 3 Dec. 15, 1998
"Logo, Math, and Beauty"

Vol. 17, No. 4 Feb. 15, 1999
"The Turtle Parties Like It's 1999"

Logo Exchange is published quarterly by the International Society for Technology in Education (ISTE), 1787 Agate St., Eugene, OR 97403-1923, USA; 800/336-5191.

ISTE members may join SIG/Logo for \$24. Dues include a subscription to *Logo Exchange*. Non ISTE member subscription rate is \$34. Add \$10 for mailing outside the USA. Send membership dues to ISTE. Add \$4.00 for processing if payment does not accompany your dues. VISA, MasterCard, and Discover accepted.

Advertising space in *Logo Exchange* is limited. Please contact ISTE's director of advertising services for space availability and details.

Logo Exchange solicits articles on all topics of interest to Logo-using educators. Submission guidelines can be obtained by contacting the editor. Opinions expressed in this publication are those of the authors and do not necessarily represent or reflect official ISTE policy.

© 1998 ISTE. All articles are copyright of ISTE unless otherwise specified. Reprint permission for nonprofit educational use can be obtained for a nominal charge through the Copyright Clearance Center, 27 Congress St., Salem, MA 01970; 508/750-8400; Fax 508/750-4470. ISTE members may apply directly to the ISTE office for free reprint permission.

POSTMASTER: Send address changes to *Logo Exchange*, ISTE, 480 Charnelton St., Eugene, OR 97401-2626 USA. Periodicals postage paid at Eugene, OR. USPS# 000-554. ISTE is a nonprofit organization with its main offices housed at the University of Oregon. ISSN# 0888-6970.

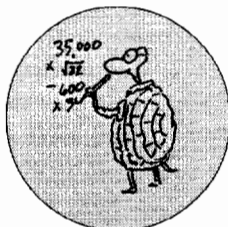
This publication was produced using Aldus PageMaker®.

Contents



ARTICLES

What I Did on My Three Summer Vacations <i>Brian Silverman</i>	5
A Mayan Pyramid <i>Orlando Mihich</i>	14
Build Your Own Virtual Pet with MicroWorlds <i>Adam Smith</i>	16
Drawing Polyhedron Patterns <i>William J. Spezeski</i>	22
Logosium '98 Report <i>Logo Exchange staff</i>	28
Logo Online	32



COLUMNS

EDITORIAL

Learning Among the Giants <i>Gary S. Stager</i>	2
--	---

QUARTERLY QUANTUM

Back to School <i>Tom Lough</i>	3
------------------------------------	---

BOOK REVIEW

The Book of Learning and Forgetting <i>Carolyn Dowling</i>	8
--	---

TEACHER FEATURE

Linda Polin <i>Gary Stager</i>	9
-----------------------------------	---

IN THEIR OWN WORDS

Surfing the Net— Rehabilitating a Metaphor <i>Jeff Richardson</i>	12
---	----

LOGO: SEARCH AND RESEARCH

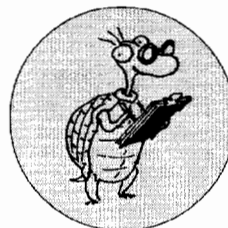
More from the President's Panel on Educational Technology <i>Douglas H. Clements and Julie Sarama</i>	20
---	----

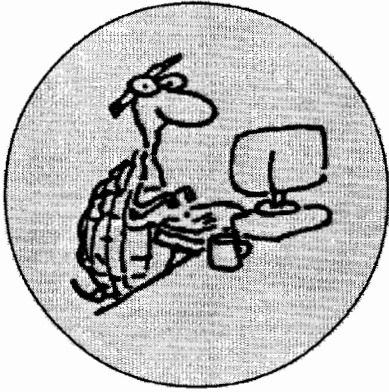
STARTING WITH STARLOGO

The Chase: Part 2 <i>Alan Epstein</i>	25
--	----

THE LAST WORD:

New Year's Resolutions <i>Susan Ohanian</i>	31
--	----





Learning Among the Giants

Welcome back! As we begin another school year it's only fitting that we take a moment to remember and celebrate our teacher heroes. Recent weeks spent teaching Logo in several Australian schools renewed my admiration for teachers (even ones who don't use Logo). Besides all of the traits we know teachers possess—poise, courage, wisdom, patience, kindness, creativity—I was reminded that teachers also require the stamina of triathletes. Teaching MicroWorlds to 120 laptop-toting seventh graders (at once) was quite an ordeal.

I started thinking about the greatness embodied by teachers recently when the world lost two giants of the profession. Jazz singer Betty Carter passed away in September. She was one of the greatest vocal improvisers and teachers who ever lived. Betty Carter felt a personal responsibility for keeping the music she loved alive and ensured just that by working with young musicians throughout her stellar career. She was committed to filling the world with music of the highest caliber.

I don't know if "Ms. BC" knew of Logo or even owned a computer. What I do know is that every Betty Carter performance was a samba school. She collaborated with her young musicians and made them better through improvisation, humor, praise, gestures, a whisper in the ear and her example. Betty Carter was respected for the teaching she did via countless clinics in public schools and an annual intensive week she led for aspiring musicians.

In the best sense possible, every performance by her ensemble was the embodiment of mutual growth, creativity, expression, and swing. I last saw the Betty Carter Quartet live in New York this past April. During the introduction of the band members, Ms. Carter said, "remember the faces of these young men because in a few years you will hear them again somewhere and think to yourself, 'My, how they have improved.'" At the end of the last set of the week, she invited one young musician after another to come up from the audience and sit in with her band. Her scat-based battle with these musical hopefuls made them dig deep within themselves and perform beyond their expectations. When the number of aspiring students began to snake around the nightclub, Betty Carter passed the torch to the next generation. She turned to the young sweating drummer and said, "It's yours, baby. Let's see how you get out of it." Betty Carter sat down in the audience and laughed out loud, visibly proud of her students.

While I was never good enough to play with Betty Carter, I was good enough to play with Richard Lukas. In fact, thousands of junior high school students in Wayne, New Jersey, had the opportunity to perform with him over three decades. Mr. Lukas was my junior high school band director. He taught me to play the trumpet and to swing a tennis racquet. He became one of my oldest friends—despite his early advice to my parents that they make my trumpet into a lamp. Tragically, Dick Lukas

died of a heart attack in September of this year at only 54 years of age.

Mr. Lukas' stage and tiny office were hothouses where less conventional students were cultivated. He was underappreciated by his superiors, misunderstood by his peers and disparaged by parents who didn't want their children to become musicians but to perform familiar ditties at two concerts per year. His ensembles always played music thought to be above the heads of students, but we were constantly rewarded by triumphing over challenging compositions. I remember the senseless controversy caused when Mr. Lukas decided to dedicate some of band time to the learning of music theory and history. He rightly believed that the school band served many purposes. Among them was for us to learn all about music, learn through music about ourselves and become well-rounded citizens.

I studied privately with Dick Lukas on and off through college and enjoyed the rare privilege of becoming my teacher's teacher when he enrolled in a series of my LogoWriter courses. My teacher/student became an avid computer enthusiast who at one point had an Apple IIgs with thousands of dollars worth of RAM, drives and interface cards. We used to joke that his IIgs supercomputer could run a small country. Over the past couple of years, he asked me about the Internet and I sought his advice regarding instruments for my junior high school children. We had come full circle. Every time my son asks

See AMONG GIANTS (Page 4)



Back to School

Ahh! The beginning of another (northern hemisphere) school year is upon us. The excitement of seeing our students once more is heady indeed! I can't wait!

In ruminating about a suitable back-to-school theme on which to ponder, the thought entered my head: Why not go straight to the fundamental idea? OK, here goes!

Question: What do you predict will happen if you type the following in your favorite Logo interpreter or command center and press Enter or Return?

back to school

Assignment: Put your Logo Exchange down, give it a thought or two first, and then try it.

As you might imagine, the response really depends on your hardware and software. (Just for fun, I will include a matching activity at the end of the column that includes a sampling of different responses. This will give some of you "ole turtles" a chance to show off a bit, and will give you "young snappers" a sense of the rich heritage from which today's Logo versions have sprung.)

I want to put enough space and text between the assignment above and what I want to say next so that you really can give it a fair try, even if your eye sort of involuntarily skipped on ahead—sort of like "inadvertently" looking at the last panel of the comics before you are really there. OK, this seems to be about the right distance.

Of all the responses, the one that was most thought-provoking to me was

I don't know how to to

What? How to to? Yep, that's what it said. Now how can we make some sense of it?

I personally feel that the procedure construct is one of the most powerful features of Logo. But as powerful as it is, have we and our students exploited its full potential? Likely not.

Well, evidently the word "to" was playing some sort of a dual role for that particular hardware and software version. It probably was not a reserved word or primitive, and yet it functioned as one when used to define a procedure. In fact, I was able to define a procedure named to! Strange. Interesting.

But wait! There's more! Let's think about what "to to" might mean to us as teachers and students returning for another school term.

One of my favorite ways of describing a procedure is that it is similar to the infinitive form of a verb. By suggesting that verbs be selected as procedure names, you can facilitate your stu-

dents slipping very easily into a multitude of interesting phrases.

**to dance
to move
to sidestep
to take one
to slide**

How to "to" might be a straightforward way for students to encounter the very concept of an infinitive for the first time.

But, "to" can represent the fundamental concept of a procedure as well. And it might be here that we can think a bit about working with our students in a particularly effective way.

I personally feel that the procedure construct is one of the most powerful features of Logo. But as powerful as it is, have we and our students exploited its full potential? Likely not.

As we and our students return for another period of time together, let's take an extra moment to think about procedures. For those students coming "back to school" who have not yet learned "how to to" it would be especially important that their introduction to procedures be crafted carefully and thoroughly.

Let's give them lots of time to play around with the idea, and lots of opportunities to try different things.

For the more experienced students, an in-depth review of procedures along with a few engaging challenges might kindle in them an interest in exploring

See BACK TO SCHOOL (Page 4)

the construct even further. Perhaps an emphasis on the idea of subprocedures within a superprocedure or maybe input variables might be something to look at in a bit more detail.

For our advanced students, the combination of input variables, conditionals, output, and recursion will always have something new and different to offer with procedures, given the right kind of stimulation. Brian Harvey's trio of advanced Logo books (MIT Press) abound in such activities and even higher level challenges, for example.

When our students come to a fuller understanding of the true power of procedures, then "how to to" will become more of a life skill and less of an error message! And that is certainly worth coming "back to school" for, isn't it? Let's make it so.

```
to into :x
output :x
end
```

```
to life
output 100
end
```

forward into life

Tom Lough, Founding Editor

PS: Here is the matching activity I promised. You may find some real surprises.

Good luck!

Instructions: Match each hardware and Logo software combination with the response you think you would get after typing "back to school" and pressing Enter or Return. Correct answers appear upside down on this page.

Hardware and Logo software combinations

- A. LogoWriter with Apple IIe and PC
- B. Logo PLUS with Macintosh
- C. PC Logo for Windows v.2 with PC

- D. MicroWorlds with Macintosh or PC
- E. Sprite Logo with Apple II+/c/e
- F. LEGO Dacta Control Lab with Macintosh or PC
- G. Terrapin Logo with Apple II+/c/e
- H. Apple Logo with Apple II+/c/e
- I. LEGO TC logo with Apple IIe
- J. PC Logo 4.0 for DOS with PC

Responses

- 1. I don't know how to back
- 2. I don't know how to
- 3. to didn't output
- 4. >
- 5. Can't use to as a command
- 6. to didn't output to back
- 7. The procedure back needs a number as its first input.

Notes

- 1. Not all hardware and Logo software combinations could be represented. For simplicity, no capitalized words are included in responses. Be aware that, for some (mostly earlier) Logo versions, capitalized words have special significance. Also, details of a change of mode and opening of an edit window (which may be associated with responses 4, 6, or 7) are not included.
- 2. Thanks to former LX editor Dorothy Fitch for her enthusiastic assistance in putting this activity together.

Answers	
J-4	E-4
I-5	D-2
H-4	C-7
G-3	B-6
F-1	A-5

**Tom Lough, Founding Editor,
Murray State University
Department of Elementary and
Secondary Education,
PO Box 9, Murray, KY 42071.
phone: 502.762.2538
fax: 502.762.2540
tom.lough@coe.murraystate.edu**

me about his mouthpiece, I will think of Mr. Lukas. I am grateful for his wisdom, humor, friendship, and guidance. He too filled my life with beautiful music.

In honor of my great teachers, I thank you for your dedication to improving the lives of children. Your daily heroics deserve much respect and appreciation. While schools become more reactionary and regressive, you dare to challenge your students and the system with Logo. Your students will remember you fondly.

Issue Notes

This issue contains ideas for classroom Logo use, provocative tales about teaching and learning, and reports from Logosium '98. Several of my Logo heroes, Brian Silverman, Tom Lough, Orlando Mihich, Bill Spezeski, and Jeff Richardson, contributed wonderful pieces, as did our remarkable contributing editors, Carolyn Dowling and Alan Epstein. Since much has been written recently about the quality of teacher education, this issue's Teacher Feature is Professor Linda Polin of Pepperdine University. She answers the question, "why Logo?" quite forcefully. Dan Kinnaman's work conflicts provided the opportunity to invite one of my favorite authors, Susan Ohanian, to write for *Logo Exchange*.

Please share your tales of classroom Logo use, research, and project ideas with *Logo Exchange* readers. The publication cannot exist without your contributions. I'm sorry for the tardiness of this issue and will work hard to have subsequent issues arrive in January, March and May. Like so much of what teachers do, editing *Logo Exchange* is a labor of love.

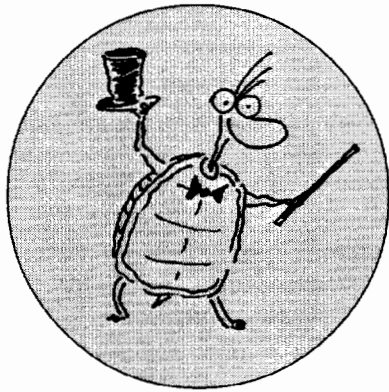
Wishing you all a year of learning and laughing.



Gary S. Stager

Gary

**Gary Stager, Editor-in-Chief
logexchange@stager.org**



FEATURE ARTICLE

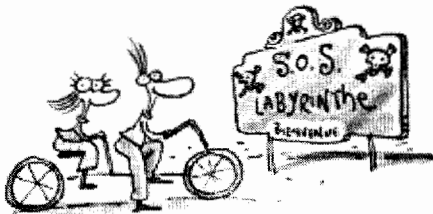
What I Did on My Three Summer Vacations

by **BRIAN SILVERMAN**

illustrated by **PETER REYNOLDS**
www.fablevision.com

Previously published in *Mathematics and Informatic Quarterly* (in Bulgaria)

We finally did it. We made it through the maze in Montreal's Old Port in 11 minutes. There's a really good chance that our time is the all-time record.

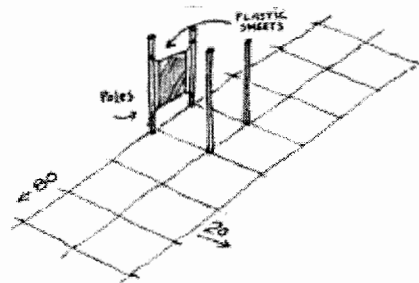


It all started a few years ago when my daughter, Diana, and I were biking and found ourselves in Montreal's Old Port. There was a new attraction called S.O.S. Labyrinthe, that promised a pirate adventure. It turned out to be a giant indoor maze in an old warehouse building with a handful of obstacles with a pirate theme. The "pirates" were kids on roller blades providing help to the desperately lost and confusion to the rest of us.



The maze is a 20-by-80 grid of about two-metre squares. The walls are made of thick plastic sheets hung between poles that are placed at the grid points. Four small sections of the maze have

been built up to resemble a ship's bridge, an engine room, a cargo hold, and lockers for the crew. These four checkpoints have hidden stampers to stamp a card received when you enter the maze. This card is also time-stamped when you first enter the maze and again when you leave.

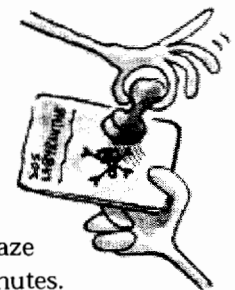


When Diana and I first tried, we got lost almost immediately. It took us about an hour and 20 minutes to find our way out and get all the stamps we needed. Despite being lost most of the time, we enjoyed it so much that we went back the following week. This time we brought my son, Eric, along because he'd missed the first time through. The second time, to avoid getting lost, we decided to follow a set of simple rules that, as any little robot will tell you, can help to get you out of most mazes.

The rules are:

- 1) turn right whenever you can
- 2) turn around when you reach a dead end.

That's all there is to it, and it actually works. We followed the rules and managed to make it through the maze in about 22 minutes.



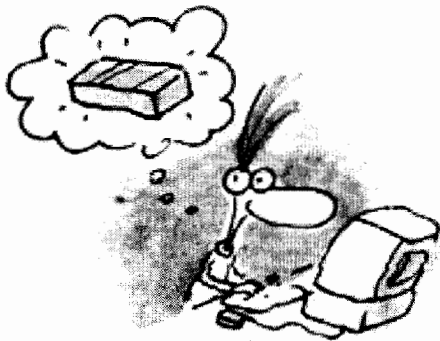
When we finished, the pirate behind the desk put our names on the board as the group that had the best time of the day. He mentioned in passing that it was a better time than he sees most days.

The challenge at this point was obvious. Our goal was to get the best time ever. We only had to figure out how. I had a plan that I thought would be pretty simple. However, as is almost always the case, it didn't turn out to be as simple as I'd initially imagined. The plan was this: Go through the maze twice. The first time through, bring along a little computer to record our path. Then go home, draw a map, find the best route and go back the following day and go through running as fast as we can.

There were a couple of immediate problems. The first one was pretty easy to resolve. How could we be sure that the maze didn't change on us between the first run and the second? (The plastic panels are moved on a regular basis to keep the maze from staying the same.) A couple of phone calls and oblique

questions later, we'd found out that the maze is changed only once a week, on Thursday night. The second immediate problem was trickier to resolve. Our plan required little computers to record our path. We didn't have any little computers. Even if we did, we wouldn't know how to make them record paths.

My friends at MIT had little computers. We'd been working for a few years on making "programmable LEGO bricks." At that time we were at the point where we'd had a couple of prototypes that had worked for a bit, but none of them were reliable enough for the task. However, as a result of a sort of spinoff of that project, there were some little computer boards around that weren't much bigger than a deck of cards. I asked my friend Randy Sargent if I could borrow one. He mailed it to me, and I had it within a few weeks. Unfortunately, by then the season was over and the project would have to wait until the next summer.



During the course of the winter a couple of things happened. One was that I had a lot of fun programming the little computer board I'd received. Over Christmas I played with making a tiny version of Logo. By New Year's we had Logo-programmable LEGO robots that didn't need to be attached to a big computer. At the same time Randy had been working on perfecting a new programmable brick. By the following summer these came together and we had a programmable brick and a logo program for saving information about where in the maze we'd been.

Little computers are pretty stupid. We would have liked to have been able

to just carry one along and have it remember where it had been. But the little computer wasn't up for the task. What we did instead was attach a couple of pushbuttons to it—one to click the number of "squares" we'd gone forward, the other to click in the amount that we'd turned at each corner.

The summer mostly slid away before we got around to trying a second run. When we did get around to it, it was just Eric and me. Before getting into the maze, we'd attached the brick to his belt and run some wires up his shirt and down the sleeves to the pushbuttons in his hands. Unless you were looking hard, you wouldn't have noticed anything suspicious. We scoped out the maze counting out loud on the straightaways and yelling out directions at the corners. People looked at us a bit strangely in general and were particularly confused and curious when we had to bring out the brick for minor adjustments.



We didn't do too well on that round. The brick started misbehaving about three quarters of the way through. And even if it hadn't, the recorded data had lots of mistakes in it. With a lot of guessing and processing we were able to construct about a quarter of the map but no more. Since it was late summer we gave up again for the year figuring we'd pick it up again the following year.

The next winter was a good one for programmable bricks. When we did the second run, there were only five work-

ing bricks in the world and even those five needed a fair amount of baby-sitting. By the next summer, there had been several new iterations of the design (largely the work of Fred Martin) resulting in dozens of working bricks that were solid enough that we wouldn't have to worry too much about hardware failures for the next round.

Also, during the winter there was plenty of time to think about what went wrong the previous summer. The main problem was that mistakes in clicking the buttons led to so much distortion in the map that it was completely useless. The maze is so big, (more than a thousand straightaways and turns), that it's impossible to do the kind of recording that we did without making mistakes. We thought a bit about eliminating mistakes but decided instead to run the experiment with several programmable bricks simultaneously, do the recording several times separately, then regroup and compare results.

As it turns out, Randy and another friend, Carl Witty, were planning to come to Montreal towards the end of the summer to show off their robots at an artificial intelligence conference. They arrived with a car full of computers, tools, and robot parts. Their robots all come with cameras connected to electronics that could discriminate colors. Their demos included robots chasing balls and each other at high speeds. It seemed only natural to get them involved in the third round.

We had a lot of discussion about whether we could use the vision systems they had in their robots for more automatic data gathering. We decided not to because even if we could resolve all the computer issues, we weren't sure that we had enough batteries for all of the needed electronics for the time it'd take. We did decide, however, that since they had brought along several miniature cameras, we'd take a video record of first trip through and use that to help interpret the data we'd get from the computers.

Carrying a camera around a maze really didn't seem subtle enough. In-

stead we took the camera and sewed it into a hat with only the lens sticking out the front. The camcorder fit neatly in a backpack. By the time we were ready to go, Carl, Randy, and I each had a programmable brick rubber banded to our belts, and Eric had a camera in his hat. The data gathering run took about two hours and was pretty boring. The bricks kept disagreeing with each other, but we ignored this because we decided to sort it all out later. Eric, originally worried that he'd attract too much attention with the camera, ended up not being able to convince anyone that he actually had one.

We brought the electronics home, dumped the data to three separate laptop computers and then spent an evening that didn't quite turn into an all-nighter trying to make some sense of it. For hours there we were, each of us with his own computer bouncing sequence numbers, grid locations, and reports of similarities and differences in data between us. My wife, Erlyne, and the kids watched for some of this and enjoyed part of the video but abandoned us when it seemed that we'd really fallen off the deep end. We persisted and after spending some time getting a feel for the method to the madness we decided to systemically play through the video noting when everything looked to be working and stopping the tape and fudging when it didn't. Our stamina ran out before the tape did, and we gave up for the night with about three quarters of the map in place.

The next morning, we all felt refreshed and raring to go. In less than two hours the printer was churning out copies of a complete maze map. We were about to set off when Eric asked why each of us had to go to get stamps at each of the places rather than split-

ting up the job. We realized pretty quickly that he had a point. There was a rule against going through the walls. There wasn't a rule against the cards with the stamps going through the walls. It took us about a half an hour of staring at the map and thinking to come up with a plan that involved three teams and three relay points to pass the cards along like a baton in a relay race. Eric and I had the first stretch, passed the cards to Randy, and then headed off to where Randy would pass them back after having met Carl twice along the way.

It all worked like clockwork. The maps were accurate, the plan workable. Eric and I had the first stamp in less than two minutes and found Randy in another two. When we called him through the plastic wall, he didn't answer but his hand appeared. He said later that a pirate was standing right beside and he was trying to not attract any attention. After

the handoff we headed to the final relay point, where we met up with Carl and got the cards through the wall from Randy. From there it was just a quick run to the end to get the last time stamp. It had taken 11 minutes, much less time than we had imagined possible.

We went to see the pirate at the desk. The board of daily winners wasn't around any more. We showed him our card that confirmed that we'd done it in 11 minutes. He said that if we did it that fast we must have cheated. Maybe it's true. Throwing that much technology at a problem may be cheating. On the other hand, it may just be another way of solving it. **LX**

About the author

Brian Silverman is the Director of Research at LCSJ where he has played a significant role in the development of MicroWorlds, LogoWriter, LEGO TC logo, IBM Logo, Apple Logo, and numerous other dialects. He is also a visiting scientist at the MIT Media Laboratory.

Brian Silverman
bss@media.mit.edu.





BOOK REVIEW

The Book of Learning and Forgetting

by **CAROLYN DOWLING**

The Book of Learning and Forgetting
by **Frank Smith**
1998, Teachers College Press,
New York
ISBN: 0-8077-7350-X (paperback)

The title of this book is irresistible. It touches on two of the core anxieties of our lives, first as children, then as adults—society expects, even demands, that we “learn,” but we know very well that what we are supposed to have learned we all too often and too easily forget! Many of us have learned extremely effectively that learning, at least within formal contexts, is hard work, and that the outcomes are at best unreliable.

Reading this book is a therapeutic experience. It is not our fault, and we are not stupid. We can all learn and do so with great efficiency and success from the moment we first draw breath. The problem, according to Smith, lies with a range of currently accepted beliefs about the nature of learning, which are actually profound misconceptions.

A large part of the book is concerned with establishing a contrast between what he terms the “Classic View” of learning (and of forgetting) and the “Official Theory.” The former acknowledges the vast amount of learning that takes place intuitively or “naturally” within social settings in which learners model their behaviors and attitudes on others with whom they identify.

Smith describes this identification in terms of joining a “club”—that is,

aligning oneself with others who possess particular qualities and competencies that, most important, we feel we will ultimately be able to achieve ourselves. Through engaging in activities cooperatively in the company of other club members, we “learn” without conscious effort.

By contrast, the Official Theory suggests that learning is the result of hard and deliberate effort within artificially contrived environments in which features such as testing, grading, and other means of contriving competition ensure that large numbers of people learn that they cannot learn—that they are effectively excluded from the “learners club.”

Smith’s Classic View of learning has much in common with the thinking of other writers and theorists including Dewey, Piaget, Vygotsky, and Papert. With such support, why does this view not prevail in schools? Smith devotes considerable space to explaining the sociopolitical as well as the intellectual foundations of the Official Theory, with an emphasis on the twin demons of psychology and of military/industrial organizational theories and practices.

Computers come out badly as well, in particular the concept of Artificial Intelligence, in the chapter titled, “The Official Theory Goes On-line.” This surprises me a little, as I recall that in an earlier work this author welcomed computing technology into the classroom, particularly as providing a rich environment for collaborative writing. In this book, his vision of the technology seems to be confined to the use of computers for individualized and dehu-

manized forms of instruction. Computers definitely belong to the “dark side.”

This is unfortunate and discouraging to those of us who experience classroom computing far more in terms of his Classic View of learning. Perhaps a complimentary subscription to *Logo Exchange* would not go amiss?

The concluding sections of the book look at the possibilities for remedying the current situation. Not surprisingly, the tone is not particularly optimistic. Nevertheless, he does provide some encouragement for teachers in regard to “Repairing the Damage”—beginning with the damage that has already been done to ourselves as learners.

Reading through this book, one finds that questions and objections inevitably arise. “But what about . . . ?” These are addressed at several points in the form of a dialogue between the author and a notional reader—an effective way of bringing broad principles “down to earth” in terms of particular situations.

In keeping with the central thesis of the book, the author suggests that we are most likely to understand, remember, and learn from it if we read it in a relaxed manner for enjoyment, rather than approaching it as a formal “learning task.”

Try it!



Carolyn Dowling
Australian Catholic University
412 Mt Alexander Rd
Ascot Vale, Victoria 3032
AUSTRALIA
c.dowling@mercy.acu.edu.au



TEACHER FEATURE

Linda Polin

by **GARY STAGER**

What is your educational background? My mother was a teacher; my father was a professor. I was doomed to end up in education. I was never officially credentialed though I've taught in high school and elementary. Several years ago I was going to get a legitimate credential but then I decided I was better not going through what you have to go through in California to get a credential. And I don't mean the tests.

I have an M.A. and a Ph.D. from UCLA in educational psychology, with a specialization in learning and instruction (i.e., instructional design and research). I worked for five years in a government-funded R&D center focused on evaluation and assessment issues. Shortly after I got my doctorate, I co-directed a five-year grant examining freshman composition at the then 19 California State University campuses. Off and on throughout all this, I was doing part-time lecturer stints at the nearby CSU campuses.

I taught research methods until the Apple giveaway in California. That occasioned the need for computer courses for teachers. I happened to have gotten into computing back at UCLA. That along with my instructional design emphasis made me a rare breed. I taught ed. tech courses for a couple of years. By then I was getting old enough to care about benefits, so I got a real job. Pepperdine had just started the first M.A. in Educational Technology, and it needed faculty. Actually they were looking for someone who could teach Logo! They were

also looking for someone who could teach research. Bingo.

Almost immediately after arriving at Pepperdine, I got involved in first of three "West Coast Logo Conferences." By the time the third one rolled around in 1989, I'd been suckered into



running it. Shortly before that, in 1986, the state of California issued the RFP for a multi-million-dollar, multi-year project to establish six California Model Technology Schools. Along with several of my Pepperdine colleagues, I participated in the grant proposal writing for a local district. We won! I had the wonderful opportunity to spend the next five years doing R&D at a local elementary school. Since it was so close to my home I spent about 50% of my daytime hours there, hanging out, sometimes team-teaching with the teachers. One of the

stories I tell below comes from this experience.

I've also done some writing. I actually wrote several chapters about Logo programming for a book that included BASIC and something else, maybe Multiplan. That was way back when there were still many versions of Logo floating around. I've also written regular columns for a couple of teaching and technology journals, including *The Computing Teacher*.

When/where did you first encounter Logo? What about it appealed to you? I have to really think about this one. I already knew about list processing when I got to Pepperdine, so it must have been when I was part-timing at Cal. State. They wanted me to teach Logo so I "obtained" a copy and sat down one weekend with one of Abelson's spiral bound books on Logo. I eventually had to learn both Terrapin and Apple Logo, so I can't recall if my first Abelson was the red one or the blue one.

I once taught a Logo programming class that had a couple of very bright programmers in it. We used to do dueling code. We'd take turns throwing out a challenge and seeing who could nail it in the least number of lines. Let's face it, programming is fun.

It was so easy to learn and I could do powerful things so quickly that I just kept going. To some extent, it's very much what Seymour wrote about early on. I dug controlling the machine. Then I dug mastering Logo. Then, to be honest, I dug the Logo people.

What is the coolest thing you've ever done with Logo? That's hard to say because I've done an awful lot of things I considered cool in their time. I recall really digging my animated adventure called Teachers in Space (after the Muppets' Pigs in Space) that I hacked out in a Terrapin Logo dialect. Then the Challenger accident happened and I erased the program. I once wrote a text-based adventure game called Get Your Masters, in LCSi Logo. You assumed an identity and had to wander around the halls of the Graduate School and interact with professors who gave you tokens. You got enough tokens you got a degree. I can't believe I spent all that time doing it, but I did. Why? Cause it was fun and I apparently had way too much free time on my hands. When LogoWriter appeared I spent some time hacking out a cool California Missions program one weekend. Father Serra walks the missions. I used to show these things to teachers and demand equivalent kinds of projects from them. They hated me; then they ended up getting obsessed themselves.

Tom Lough used to sell me discounted Lego Logo sets used in demos. I hoarded them in my home until my then-8-year-old kid and I mastered some serious programming efforts. Then for a few semesters I made my students build and program their brains out. We had great fun, lots of crawling around on the floor and cutting up cardboard.

Can you share your most interesting teaching with Logo story? I have two favorites, but I'll only tell one. This is a long story but a cool one. I was hanging out in a technology-rich elementary school as an ethnographic researcher in a participant-observer role. Everyone knew I knew a lot about computers so they asked me questions. Two teachers approached me in the lounge one day at recess. They were going to do a cross-age project. It was of limited vision in classic ways. The fourth grade was going to sit with the

second grade kids and run a Logo program about Autumn and then keyboard in the stories the second graders came up with that were supposed to go along with the program. They wanted me to write the program for them. They wanted a tree with leaves that would fall. Between recess and lunch I hacked. I had to do stuff like create the different shapes of the leaves falling and toss in some variables to alter the trajectories so they didn't just stamp on top of each other. Got the thing done and showed it to them. They liked it. We discuss how we ought to introduce it to the fourth graders so they would understand how to work with the procedures. I said I'd team with the fourth grade teacher to do this.

"My relationship with Logo made me very cynical about educational software. It was awfully darned hard to find anything as good for the student as Logo. It still is."

I showed the program to the fourth graders several times and talked about the fact that you could vary the path the leaves took to the ground. I wanted them to understand the variables. I wrote the key lines on the board: repeat X times a list that had the turtle set heading, set a shape, move along a trajectory, set to a different shape, move further. Then it stamped the leaf shape. Initially the heading was 180, straight down. I ran the program several more times and the kids noted that the leaves were all stamping on top of each other. They wanted to have the leaves distributed at the base of the tree. Heh, heh.

I pointed to the line of code, explained it in English, and asked what we should change. They suggested altering the heading to something like

190 degrees. I did so in the program and ran it again. The leaf didn't make it to the ground. It stamped just above the ground level. The kids were confused. We did this a number of times. I suggested that we walk through the action in the line. As we acted it out with student-leaves repeatedly walking, turning, changing shape, the kids got the idea of changing the distance that the leaf went. Aha, I thought, they got it.

They suggested shortening the distance the leaf traveled. This mystified me but I let them do it. The leaf stopped even further from the ground. I looked at the teacher. She didn't get it either. I let them tinker a bit more and when it was clear they didn't have an alternative plan in mind but were just hacking I suggested we act it out again. This time we cleared the entire tile floor and we drew the tree on the floor with chalk (God bless tile linoleum). We placed the student leaf on the tree and walked through the changes they'd suggested. Sure enough, the student leaf stopped short of the ground level of the tree.

After they ran the human demo a couple of times I asked them why they thought the leaf should travel a shorter distance to the ground when they changed the heading. Get this. They said, "it's a short cut." I didn't understand what they meant. "What's a short cut? Where's the short cut?" Through much excited discussion we eventually had a meeting of the minds. They saw the widening angle of descent as a triangle alright, but they mapped that onto their personal experience with triangles...as short cuts across the grassy corner of right angle sidewalks. They experienced the hypotenuse as a short cut.

Now that we had some common ideas to talk about it was time to return to the problem of the leaf path. If cutting corners was a short cut, how come the leaf had to travel further? They didn't connect the increasing angle with lengthening of the hypotenuse. I looked over the kids' heads at

the teacher and said, "Well we're on the brink of the Pythagorean Theorem, shall we go for it?" I mean, we even had square tiles on the floor we could have used for the measurement. She didn't want to, but I think she saw, for the first time, what this Logo stuff was all about.

By the way, the leaf marched forward until she reached "the ground" and we counted up the tiles. We put the leaf-kid back at the start; changed the trajectory and did it again. We recorded our floor tiles and eventually they began to see the the spread of the angle having a direct effect on the distance needed to get to the ground.

What has been Pepperdine's relationship with Logo? Four faculty members traveled back to visit with Papert and the Hennigan School back around 1985. That was fairly early in our relationship with the Logo community. Pepperdine's M.A. in Ed Tech began in 1983 and it had a Logo course in it then. We ran a West Coast Logo Conference in 1985, 1987, and 1989, though by 1989 the conference had broadened to include some other products. I got there in 1985, in time for the Hennigan visit and the Logo Conference.

We've always supported Logo as an important piece of our teacher credential and MA programs, though it's no longer a class unto itself in the ed. tech. program. I think we began with Sprite Logo on Commodore 64s and our first Logo TC logo version on the Apple //e.

How has Logo influenced your thinking about education and educational computing? My relationship with Logo made me very cynical about educational software. It was awfully darned hard to find anything as good for the student as Logo. It still is. I suppose this isn't a good thing to say in print, but well, a lot of educational software is evil. The developers just don't get it. They're working too hard to manipulate everything and wedge the student down some prepared pathway to knowledge. It should be about think-

ing, not remembering. Logo, in social constructionist hands, is marvelously expansive. This can be scary to people who think teaching and learning is about moving along cleverly laid pathways. I think Peter Reynolds book, *The North Star*, explains this very nicely.

The Logo Disaster (my phrase) taught me a lot, too. I'm referring to the research backlash of the late '80s that condemned Logo as failing to deliver on academic promises. The research world really screwed up on this one, and today I still get asked to sit on panels to discuss the measurement of effects of computer use on kids' learning. (Hopefully, I don't have to launch into a harangue on the absurdity of this issue for this audience.) Anyhow, Papert wrote a terrific rejoinder which was in fact published in the right place for such a response (*Educational Researcher*, a journal of the American Educational Research Association), but the damage was done. I was deeply impressed by the impact that had on the Logo "movement." It scared the heck out of me and was an early wake-up call to me as a technologist (is that a word?). However, Logo was also just at the wrong place and the wrong time.

Pepperdine is an obvious leader in graduate-level teacher development. Why does a university with the first online graduate degree programs focused on emerging technologies, community and constructionism still teach MicroWorlds? 'Cause we're smart. We don't just teach this stuff, we get it.

Why is it important for every perspective teacher in a credentialing program to experience learning with Logo? Lots of reasons, mostly insidious. I've discovered teacher credentialing candidates tend to have very brittle notions of teaching and learning. Despite taking psych or curriculum courses, when push comes to shove they revert to their true beliefs about teaching and learning. Those beliefs are largely behaviorist, most likely because that's

what they experienced as students." So we get them back in their student mode and let them reexperience their own education in a more enlightened fashion. Microworlds is a big part of that. It lets us arrange the kind of activities/tasks that encourage play, interaction, and goal-directed inquiry. I think Hesse said "he who would be born must first destroy a world."

The other good reason is to offer them some insights about mathematics. Many elementary teachers do not have a deep grasp of what they're teaching in mathematics. They aren't members of the math culture. They don't have mathematical ideas. They don't think mathematically. As a result their math instruction tends to lean on procedural knowledge, and that tends to lead them toward practice and memorization as a method. Now, we can't remake their math educations but we can crack open the door to some interesting insights that might lead them to recognize the difference between what they know and what math is, and they might be encouraged to do something about it. I suppose that might be possible with programs like Geometer's Sketchpad and Tabletop, but those are overtly math-y programs. The beauty of Logo is that the math is naturally embedded in the doing of something else. It isn't only or explicitly about math. It's deliciously tacit. **LX**

About the teacher

Dr. Linda Polin is a professor and program director at Pepperdine University's Graduate School of Education and Psychology. Check out Pepperdine's Online Program Web site at <http://gsep.pepperdine.edu/online/> or Linda's personal Web site at <http://moon.pepperdine.edu/~lpolin/>.

Linda Polin
lpolin@pepperdine.edu

Surfing the Net— Rehabilitating a Metaphor

The 1998 Logosium Keynote Address

by JEFF RICHARDSON



I'm sick of hearing and reading this phrase, sick of seeing images of computing paraphernalia balanced on surfboards and breaking waves. It's not that I don't like surfing though. I love it.

Surfing is exhilarating. Surfing can be attempted by a beginner on a gentle shorebreak. And surfing offers endless challenges to the fittest and most creative. Surfing can be dangerous, and requires a high level of skill, knowledge and fitness to do well. Surfing culture has a great sense of community and shared knowledge and expertise. And in Australia, where I live, safety in the surf is provided by a wonderful tradition of Surf Lifesaving Clubs. These are volunteer community organizations that patrol popular beaches throughout the summer season. They effect rescues, provide first-aid, and induct children and young people into surf culture and lore through organized competition as well as the less formal, individualistic approach that is emblematic of surfing.

So why do I get annoyed at surfing being used as a metaphor for browsing the World Wide Web? And what's this got to do with Logo?

The metaphor of surfing migrated across to Web browsing from television viewing. Surfing was a bad metaphor for lying on a couch pressing a TV re-

mote control, and it's a bad metaphor for pointing and clicking with Netscape. Both these things have none of the attributes of surfing that I've listed. And they both pick up on the 1960s mainstream stereotype of surfing as a "teen" cultural phenomenon of "cheap thrills and a free ride."

The metaphor of surfing migrated across to Web browsing from television viewing. Surfing was a bad metaphor for lying on a couch pressing a TV remote control, and it's a bad metaphor for pointing and clicking with Netscape. Both these things have none of the attributes of surfing that I've listed.

Now this does a disservice to surfing, and that annoys me for a start. You can find yourself being urged to "surf on over to http whatever," as though what's going on when you're using all your strength, wits, and wisdom to catch a two-metre wave breaking over

an offshore reef is something akin to riding public transport. But in invests Web browsing with a glamour that it just doesn't deserve. There are ways of using a computer that are much more akin to surfing, and one of the best I know is that offered by Logo and the Logo Community.

Surfing is above all active rather than passive. And I think this is what Brian Harvey was driving at in his article "Stop the Internet I Want to Get Off" (*Logo Exchange*, Volume 16 Number 1 Page 18). In the article, Brian distinguishes the World Wide Web from other uses of the Internet by exposing it as a publishing rather than a conversational medium. While I agree with that distinction, I want to argue for the appropriation of the World Wide Web, or at least a part of it, by children. And I can think of no better way to do this than by using Logo, and in particular, MicroWorlds. MicroWorlds used in combination with the MicroWorlds Web Plug-in is truly "Java-for-kids". It's capable of much, much more than the mundane hypertext of a typical Web site. Already, some childrens' work is beginning to appear as MicroWorlds Web sites. For a sample, I recommend to you <http://el.www.media.mit.edu/logo-foundation/VGW/>. This site, facilitated by The Logo Foundation, has video games, playable over the

World Wide Web, that were made by Manhattan schoolchildren.

The Logo idea is not just about programming the computer. It's about computer programming as an expressive medium, and it's about using this expressive medium in a community. A Logo community can be a microcosm, like a classroom or, now with the Internet, the Logo community can and does circle the earth. Making things and doing things with Logo is a lot like surfing. You take risks. You crash and bump and learn from your mistakes. And you learn from others, from watching others, and from asking others. You get exhilarated by what you achieve and infuriated by what you can't achieve. But you don't give up. You dive back in and have another go. And you have a community of peers and elders who you can turn to, or who will help you out if you get into bother. You can do things your way, you can be creative. There's no single correct approach to things. All this is a lot like surfing and surfing culture.

The Logo idea has a lot in common with other progressive philosophies of education, one of which is the "process" or "whole-language" approach to Language Arts. A key element of a process approach to childrens' writing is publication, and with it a sense of audience. Publication can take many and varied forms, and audiences can be large or small. But the notions of publication and audience bring many levels of richness and function to writing and the learning of writing: a sense of purpose, ideas of genre, and issues of presentation and orthography are but a few. I think that publishing Logo projects using the World Wide Web offers all these extra dimensions to the

Logo idea. The publication of childrens work on the World Wide Web can be as limited and semi-private as you wish, or completely unrestricted. It can allow for feedback and communication, or be relatively anonymous. Choosing how to do this is a much more authentic en-

The Logo idea is not just about programming the computer. It's about computer programming as an expressive medium, and it's about using this expressive medium in a community. A Logo community can be a microcosm, like a classroom, or now, with the Internet, the Logo community can and does circle the earth.

agement for educators with the issues of censorship and safety for children on the Internet than all the time and energy that is currently devoted to "netnannying," "Web-whacking," and "acceptable use policies."

Dan Kinnaman, regular columnist in *Logo Exchange*, made the earlier observation (I'm paraphrasing Dan's words from memory, so my thanks and apologies to Dan) that we hear a lot about connecting the Internet to classrooms but never about connecting classrooms and children to the Internet. Which way is it going to be?

Which way should it be? And which will be most changed by the connection, the classroom or the Internet?

There's a saying used by old hands in the Surf Lifesaving movement that "when a new bunch of youngsters join the Surf Club, you line them up against the wall and throw a housebrick at them. The ones who duck out of the way, they're the smart ones; you train them for beach patrol. The ones who try to catch the brick, they're competitive, so you put them in the iron-man and the surf races. The ones who just stand there and let the bricks hit them, you put in the whale boats to row out through the surf." We're a very diverse bunch in the Logo community, so as we start surfing with Logo on the World Wide Web, I think there'll be a role for each of us. I'm not going to throw any bricks, so it's up to you to wax up your Logo, paddle out, and start surfing.

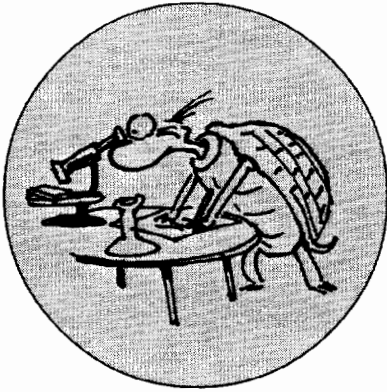
I'm sure that next year's Logosium will see a proliferation of child-constructed Logo Web sites. I'll thank you now in advance for your part in facilitating them.



About the author

Jeff Richardson is International Editor of *Logo Exchange* and a Senior Lecturer in Educational Computing at Australia's Monash University. You can look at one of his MicroWorlds projects at www-mugc.cc.monash.edu.au/~jeffr/Archimedes.html (He thought he should put his Logo code where his mouth is!)

Jeff Richardson
jeff@rmit.edu.au



A Mayan Pyramid

by **ORLANDO MIHICH**

This school year my sixth-grade students worked on several Logo projects. One was the building of the famous Maya pyramid, El Castillo, at Chichen Itza, Yucatan, Mexico, using nothing but rectangle shapes.

The idea of drawing Mayan pyramids began about two years ago when my students and I followed the MayaQuest expedition in Central America. It was a great experience for us to read the daily reports, and meet all those people along the Ruta Maya. Unfortunately, the expedition has been abruptly terminated this year, but the rich diaries, interviews, and events are still online for students to explore. It was then that one of my students made a drawing of a pyramid, and it was an inspiration for other students to draw additional pyramids.

This year I photocopied a blueprint of the pyramid with some dimensions in turtle steps. Students had to do some math to figure out additional values themselves and to invent some solutions.

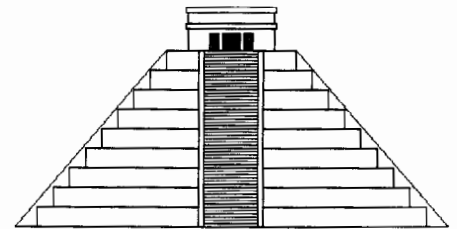
Students had a lot of fun creating the 54 steps of the frontal staircase with the repeat function, and even more fun using the recursive procedure to build the sides of the pyramid. They proved very creative in reversing some of the procedures, or creating new ones (e.g. left.rectangle). At the end of the

projects, some students used code to paint their pyramids, others followed, and very few used the MicroWorlds paint tools. "It was too easy," they said. In addition, some students created the three Mayan glyphs for pa, ka, and la, which stand for the name of the Mayan king Pakal.

Next year I plan to expand the project by adding some more Logo pyramids from the splendid Maya civilization, other ancient sites like Palenque, Copan, and Tikal, "the place where the count of days was kept." We hope to create a Logo Mayan calculator as an approach to the mathematics of the Maya.

Basically, this is an old dream. Years ago I traveled extensively throughout the Maya world, intrigued by the mystery and beauty of these ancient sites. After this year's encounter with the Logo pyramid, my students will definitely recall the name Chichen Itza, pronounced in class so many times. I hope, as time goes by, they also remember some of the words of Ms. Linda Schele stressing that, "American history does not begin in 1492 with Christopher Columbus, it begins with the first Maya king who wrote his name on the stone, in 200 BC."

Here is the MicroWorlds (Logo) code created by one of my students, Charlie Lu:



```
to start
pu
home
bk 80
lt 90
fd 25
rt 90
pd
end
```

```
to rec :h :l
repeat 2[fd :h rt 90 fd :l rt 90]
end
```

```
to staircase
repeat 54[rec 3 50 fd 3]
rt 90 bk 5
rec 5 54 * 3
fd 55
rec 5 54 * 3
fd 5
lt 90
bk 54 * 3
end
```

```
to build.r :h :l
if :l < 20 [stop]
rec :h :l
```



```

fd :h
build.r :h :l - 15
end

to recl :h :l
repeat 2[fd :h lt 90 fd :l lt 90]
end
to build.l :h :l
if :l < 20 [stop]
recl :h :l
fd :h
build.l :h :l - 15
end

to sides
build.r 54 / 3 150
bk 54 * 3
lt 90
fd 60
rt 90
build.l 54 / 3 150
end

to temple
rt 90
bk 10
lt 90
rec 20 80
fd 20
rt 90
bk 2
lt 90
rec 5 84
fd 5
rt 90
fd 2
lt 90
rec 10 80
fd 10
rt 90
bk 2
lt 90
rec 5 84
rt 90
fd 2
lt 90
pu bk 35
rt 90
end

to entrance
fd 20
lt 90 pd rec 16 8
rt 90
fd 12
lt 90 rec 16 16

```

```

rt 90
fd 20
lt 90 rec 16 8
fd 8
rt 90
pu fd 4
setc 7
repeat 3[ pd fill pu bk 16]
setc 9
end

to staircase.l.r
pu start
lt 90
fd 155
pd fd 20
make "a pos
bk 175
rt 90
fd 54 * 3
lt 90
fd 35
setpos :a
bk 20 + 150 + 10 + 50 + 150 + 20
make "b pos
fd 170
rt 90
fd 54 * 3
rt 90
fd 30
setpos :b
end

to startup
cg ht
start
staircase
sides
temple
entrance
staircase.l.r
end

```

About the Author

Orlando Mihich, a graduate chemical engineer, has been teaching Logo and Science to middle school students for the past 14 years. He is currently a computer teacher at Booker T. Washington MS 54 in New York City, and an instructor at the New York Institute of Technology.

Orlando Mihich
OMihich@aol.com

RESOLUTIONS / Continued from Page 32

dard curriculum to students, then we'd be better off with quill pens. It is both the opportunity and the obligation of teachers using computers to support idiosyncrasy.

Resist much

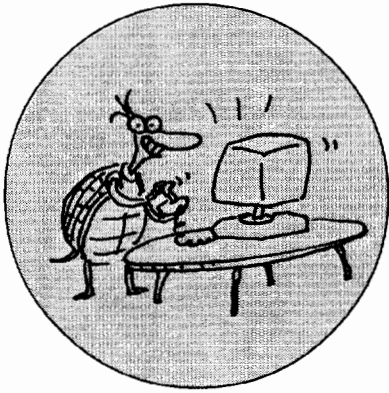
Even such a supreme rationalist as George Bernard Shaw confessed to a fascination with machines, saying he once nearly bought a cash register "without having the slightest use for it." We teachers are probably more vulnerable to things mechanical than most. We live harried, hassled lives beset by hucksters and inspectors alike. But we must keep track of what matters, what the children need. We must decide what computers should do, not merely celebrate what they can do. We must not allow sharpshooting wheeler-dealers to bring in their millennium-skills promises, turning our classrooms into techno-skill dumps, hazardous to the well-being of children. We must question. We must judge. And always, always, we must resist much. **LX**

About the author

Guest columnist Susan Ohanian is the author of hundreds of articles and the author of popular books on teaching and learning including: *Garbage Pizza*, *Patchwork Quilts and Math Magic*, *Who's in Charge? A Teacher Speaks Her Mind and Ask Ms. Class*.

Last Word Contributing Editor Daniel Kinnaman was unable to meet even the elastic deadlines for this issue of *Logo Exchange*, since he has recently been named publisher of *Curriculum Administrator Magazine*. We wish Dan well in his new role and look forward to him appearing in future issues of *Logo Exchange*.

Susan Ohanian
ohanian@stager.org



Build Your Own Virtual Pet with MicroWorlds

by **ADAM SMITH** with a bit of help from **GARY STAGER**

Editor's Note

In 1997, millions of little digital computers costing less than \$20 were purchased by children around the world. These virtual pets were connected to key chains and known as Giga-Pets, Nano-pets, Tamagotchis and variations on the terms digital, personal, virtual or pet. Kids fell in love with controlling and caring for these pets. The predictable response of most schools was to ban them. (There seems to be a causal relationship between things kids love and things school bans. Remember what a terrible threat POGS or baseball cards posed to contemporary education?)

Thoughtful educators realized that these digital pets provided children with a personal connection to systems thinking, senses, emotions, artificial life, computer science, cybernetics, probability, elapsed time, human development and other complex concepts. While most schools were prohibiting these "pets," Logo teachers around the world were independently developing Logo-based activities which used digital pets as a powerful motivational force for learning. Adam Smith, an Australian educator and software developer (www.schoolkit.com) devised a simple set of starter procedures for creating a digital pet. The project introduces buttons and text boxes. Students with little MicroWorlds programming experience may successfully create a digital pet and add their own personality traits to their creature.

I've added a few extension ideas to what Adam created.

What is a Personal Pet?

Your Personal Pet can be anything you would like it to be, a dog, a horse . . . or something imaginary. (You may simulate all sorts of natural systems this way. Josie Hopkins, a teacher recently featured in *Logo Exchange*, had her students make Tamaseedis — virtual plants to care for.)



Say 'Hi' to Stroops, my Personal Pet!

Caring for Your Personal Pet

Like all pets, Stroops has needs and feelings. When you adopt your pet, it will get hungry, so you will need to feed it. To keep it happy, you will need to hug it regularly.

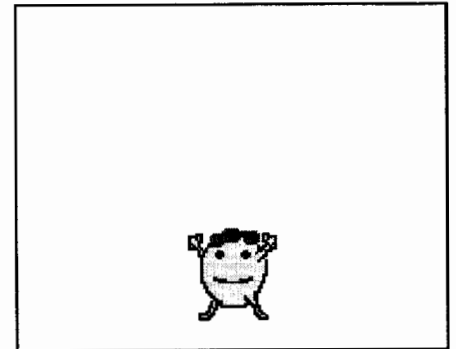
Once you have adopted a pet, you may like to add more characteristics. For example, you might like to have your pet become lonely unless you play games with it.

Adopting Your Personal Pet

To adopt a Personal Pet you need to "bring it to life" by writing a MicroWorlds program. As you work on your pet, be sure to save your project regularly.

Draw a New Pet

Use the Shapes Center to change the shape of your turtle into a picture of a Personal Pet. You may want to draw your own shape, or use one of the shapes provided. When you are done creating your new shape, put that shape on the turtle. Your screen should now look something like this:

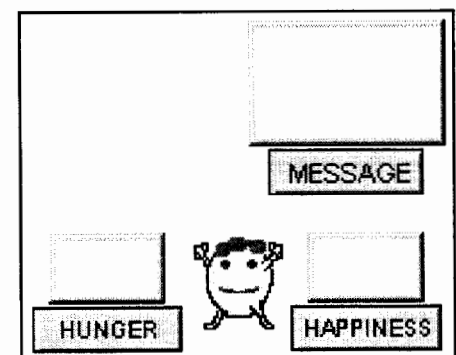


Add Text Boxes

Put three text boxes on the screen so that you can see what the pet is thinking and feeling. Name the text box objects:

- **hunger**—This text box will display a number to show how hungry your pet is.
- **happiness**—This text box will display a number to show how happy your pet is.
- **message**—The pet *may* use this text box to display messages.

Your screen should now look something like this:





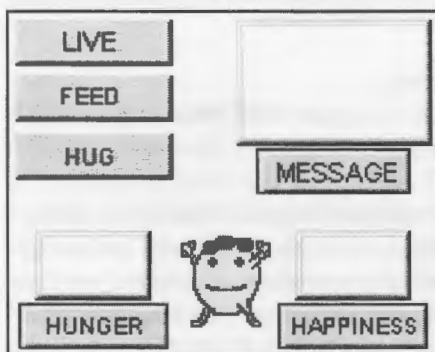
A more advanced virtual pet by one of Adam's sixth graders

Add Buttons

Put three buttons on the screen so that you can interact with the pet. Each button should be set to ONCE. Name the button objects:

- **live**—This button will start the Personal Pet program.
- **feed**—Click on this button to feed the Personal Pet so that it is not hungry.
- **hug**—Click on this button to hug the Personal Pet to make it happy.

Your screen should now look something like this:



Enter the Procedures

Now it is time to program the Personal Pet. On the Procedures Page enter the procedures written below.

MicroWorlds text boxes may be used as visual variables. They display their values constantly and may be changed and operated upon under program control. A text box, *HUNGER*, has a corresponding one-input command, *SETHUNGER*, designed to change its contents.

We will use the *SETUP* procedure to set starting values. For example, the Personal Pet begins with a Hunger level of 20.

```
to setup
message, ct
sethunger 20
end
```

In *HOUR* we will put everything we want to happen each time a "pet hour" passes. For example, each hour the Personal Pet becomes a little hungrier.

```
to hour
message, ct
sethunger hunger + 1
if hunger > 40 [announce [I
died from hunger!] stop]
end
```

When you click on the *FEED* button, the Personal Pet becomes less hungry.

```
to feed
if hunger > 4 [sethunger
hunger - 5]
end
```



Stroops loves eating!

The *LIVE* procedure watches the time, making sure the *HOUR* procedure is run each "pet hour."

Question: Why do "pet hours" go by so quickly?

Answer: For the purposes of debugging, testing and program design you don't want to wait three days to determine if your program works correctly. Therefore, "we'll speed up time." Once your program works to your satisfaction, you might wish to change the hour to take longer than the six seconds specified in the *LIVE* procedure.

```
to live
setup
forever [if timer > 60
[reset HOUR]]
end
```

Test the Procedures

Now go back to the main MicroWorlds page to test the Personal Pet. Try the following:

- 1) Click on the *LIVE* button. The Hunger level of your pet should go up slowly.
- 2) Click on the *FEED* button. With food the Hunger level should go down.
- 3) Don't feed the pet. Eventually it should die from hunger.

Read through the program a few times, making sure you understand how it works.

Add to the Procedures

With the basic program working, it is time to extend the program. Go to the Procedures Page and add to the procedures in the ways detailed below.

Let's add Happiness to the pet. In the SETUP procedure, add the line shown here in italics.

```
to setup
message, ct
sethunger 20
sethappiness 10
end
```

The values set in SETUP were chosen arbitrarily based on their ability to make the simulation run well. The number (40) used to check the hunger of the pet in the HOUR procedure may be is also arbitrary. If you don't like the way the virtual pet behaves, feel free to adjust the values set in SETUP or HOUR.

Each HOUR, without hugs, the pet gets lonely and a little less happy. Add the line in italics.

```
to hour
message, ct
sethunger hunger + 1
if hunger > 40 [announce [ I
died from hunger!] stop]
sethappiness happiness - 1
end
```

Notice that we represent getting unhappy by subtracting 1 from the happiness variable. We need to be able to hug the Personal Pet to keep it happy. Add this procedure.

```
to hug
sethappiness happiness + 5
end
```

Check and Extend

Now go back to the main page to test the Personal Pet. When you are sure it is working properly, you could add another characteristic such as education or fitness. You may also add a sound to the live, feed or hug buttons to provide aural feedback for your actions. Alarm sounds might be added to alert the user to unhappiness or starvation.

Since the HOUR procedure checks the condition of the pet, you should put new conditionals there. Checking for

an unhappy little critter might look something like this. The new lines are in italics.

```
to hour
message, ct
sethunger hunger + 1
if hunger > 40 [announce
[Your pet died of
starvation!] stop]
if happiness < 7 [announce
[I'm bored!] stop]
if happiness < 5 [announce
[Nobody loves me!] stop]
if happiness < 1 [announce
[Your pet died from
loneliness!] stop]
sethappiness happiness - 1
end
```

Add to the Procedures

In life, all sorts of unforeseen crises or blessings may surprise us. Sometimes, things might happen which will affect your Personal Pet. Add this procedure:

```
to event
run pick
[
[ announce [It's my birthday.
Lots of fun.] sethappiness
happiness + 20]
[ announce [I fell over and
hurt myself.] sethappiness
happiness - 10]
]
end
```

The EVENT procedure picks a list of instructions randomly from a larger list and then runs that instruction.

We need an EVENT to happen randomly. Before the end of the HOUR procedure add the following line. This change makes the EVENT procedure run based on a probability of one out of 12 times. (You may of course alter the odds.)

```
if 1 = random 12 [event]
```

Check and Extend

Now it's time to test the procedures again. Occasionally, your pet should announce a special event.

When you are sure the program is working properly, you might want to add some more events of your own into the EVENT procedure. Perhaps your pet can get ill or learn.

An Active Pet!

You now have a basic Personal Pet. There are a number of ways you can make your pet more interesting and active. If you are looking for ideas, think about all of the things a real pet does.

Link Characteristics and Events

Add more characteristics and more events and link them together. For example, you could make it rain occasionally. If the pet is not inside some shelter, it could get cold and unhappy, or even sick.

Animation

Add animation to show such things as your pet eating its food or going for a walk.

Grow-up

Design different shapes representing your pet as a baby, child, teen-ager and adult. Keep track of the number of years (some combination of elapsed "pet hours"). At a particular age, use SETSH to change the appearance of the pet. This will give the appearance of growing-up! You can even use some variation of SETSIZE SIZE + 1 in the intervening years to cause the pet to grow while aging.

Games

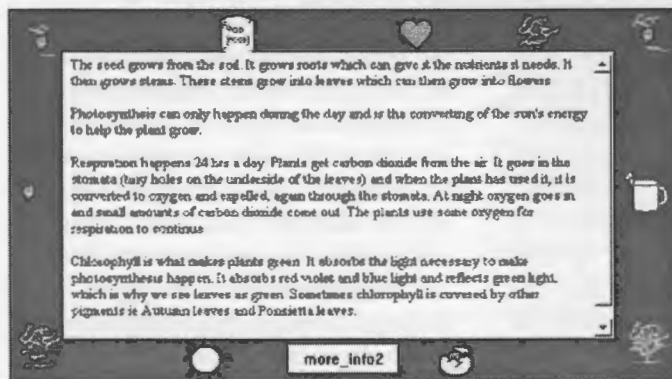
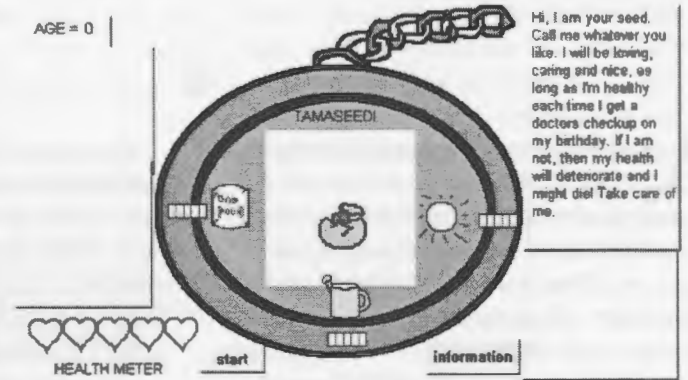
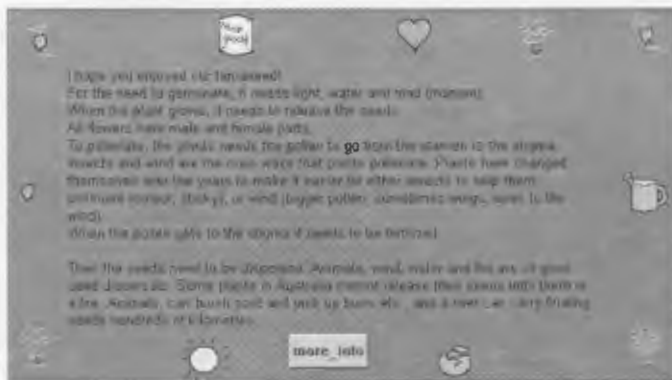
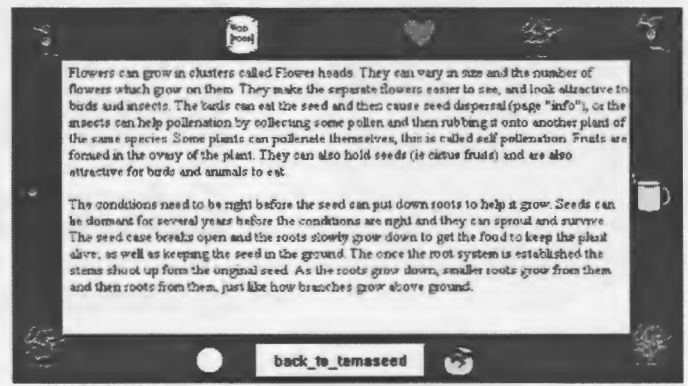
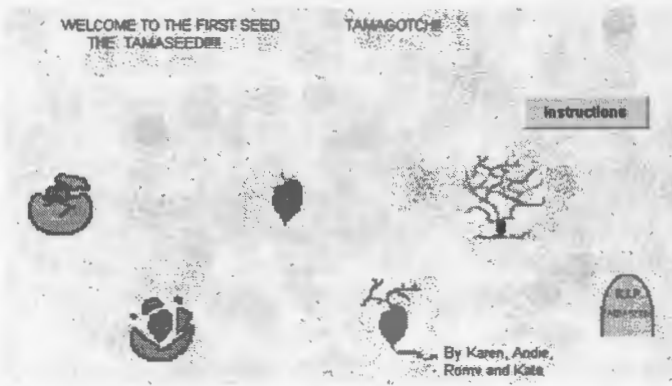
Invent a game that you can play with your pet.

Send Your Virtual Pets Out to Stud!

Please share the virtual pets created by you and your students with *Logo Exchange*. We will publish cool programs and screen shots in future issues. *Logo Exchange* would be happy to publish some of these pets on our Web server too.

Send projects and screen shots to logoexchange@stager.org.

What Kind of Virtual Pet Can You Create?



About the author

Adam Smith is well known as a veteran classroom teacher, administrator and Logo user across Australia. He recently created the Windows-based software package, *Schoolkit*, an expanding library of creative activities designed to help teachers and students use open-ended software tools like MicroWorlds and Microsoft Office. A special free sample version of *Schoolkit* is available to *Logo Exchange* readers at <http://www.stager.org/schoolkit>.

Adam Smith
www.schoolkit.com





More from the President's Panel on Educational Technology

by **DOUGLAS H. CLEMENTS AND JULIE SARAMA**

An overwhelming response from the Readers of this column¹ has convinced us to tell more about the report by the President's Committee of Advisors on Science and Technology (President's Committee of Advisors on Science and Technology—Panel on Educational Technology, 1997). In our last column, we emphasized their statements concerning how computers are used within our diverse multicultural society. Here, we provide additional highlights from that report. The panel made the following recommendations:

1. Focus on learning with technology, not about technology. Both are worthy of attention, but to realize its full potential, technology needs to be integrated with the learning of other subject matter.
2. Emphasize content and pedagogy, and not just hardware. We need wide-spread availability of technology, of course. The ratio of students to computers changed from 125:1 in 1983-84 to 10.5:1 in 1994-95. This still falls short of four or five students per computer that many experts consider reasonable. Middle and junior high schools have less access than senior high schools and elementary schools have an even higher ratio.

Even more important than access to hardware is achieving the goals of current educational reform through the use of new methods to teaching higher-order reasoning and problem-solving skills. How should we teach? The panel recommends a constructivist paradigm. They say they can not make a confident and definitive statement regarding the superiority, but "the student-centered constructivist paradigm may ultimately offer the most fertile ground for the application of technology to education" (p. 35). They include "acquisition of computer programming skills" as part of that approach.

3. Give special attention to professional development. About 15% of available funds is budgeted to helping teachers develop professionally. The panel recommends that this figure should be 30%. In addition, schools of education should be helped to incorporate technology within their curricula so they are capable of preparing the next generation of teachers. Over 200,000 teachers enter the profession each year. There is a 50% turnover in the teaching forces approximately every 15 years. Colleges of education have

to introduce future teachers to the use of educational technology before the demands of an actual teaching position begin to impinge on the time available for such training.

As a side note, I (Doug) remember when people argued that you did not need courses on computer in education because the next generation of aspiring teachers would know so much about computers. I disagreed then and I disagree now. First, not everyone has equal (and certainly not sufficient) access and experience, as we saw in the previous column. Second, knowing how to use a computer in any old way and knowing how to teach with computers are quite different.

Teachers also need help in the schools. Fewer than 5% of all schools have full-time computer coordinators capable of providing such sustained assistance. These coordinators typically spend 20% of their time helping teachers, selecting software, or formulating technology-oriented lesson plans. Teachers need someone available to them to help not only with technology, but also with challenging applications of it such as organizing projects.

4. Engage in realistic budgeting. We need more funding for educational technology over all. At least 5%, or \$13 billion should be earmarked, up from a mere 1.3% today. The report makes an intriguing point that it is not necessarily a good idea to get older computers from businesses. First, the cost of upgrading and maintaining old machines may be higher than the cost of buying new ones. Second, after accounting for loss of public revenue attributable to tax deductions claimed by the businesses, there is less money available for computers in education! The net value of corporate equipment donation may be negative.

5. Ensure equitable, universal access. We discussed these issues in the previous column.

6. Initiate a major program of experimental research. In education generally, and in educational technology in particular, funding for research is, frankly, pathetic. Compare these figures: 23% of pharmaceutical funding goes to research. However, less than .1% (that is one-tenth of 1%) of funds for elementary and secondary education are invested to see which techniques actually work. The panel recommends that this should be .5% at least, or \$1.5 billion annually. We believe this is a minimum amount. Such research should include.


a. Basic research in various learning-related disciplines and on various educationally relevant technologies.

b. Early-stage research aimed at developing new forms of educational software, content, and technology-enabled teaching. The Panel points out that it is unlike to have such work conducted without public support.

c. Rigorous, well-controlled, peer-reviewed, large-scale empiri-

cal studies designed to determine which educational approaches are in fact most effective in practice.

The report concludes that “. . . the Panel believes that such technologies have the potential to transform our schools in important ways, and finds ample (albeit partially anecdotal) justification for the immediate and widespread incorporation of such technologies within all of our nation’s elementary and secondary schools” (p. 113).

These recommendations are consistent with those frequently heard in the Logo community. This report may give us the support for and “official confirmation” of our aims and our vision. 

References

President’s Committee of Advisors on Science and Technology—Panel on Educational Technology. (1997). *Report to the president on the use of technology to strengthen K-12 education in the United States*. Washington, DC: Author.

Time to prepare this material was partially provided by a National Science Foundation Research Grant #ESI-9730804, “Building Blocks—Foundations for Mathematical Thinking, Pre-Kindergarten to Grade 2: Research-based Materials Development.” Any opinions, findings, and conclusions or recommendations expressed in this publication are those of the authors and do not necessarily reflect the views of the National Science Foundation.

About the Authors

Douglas H. Clements, Professor at the State University of New York at Buffalo, has studied the use of Logo environments in developing children’s creative, mathematics, metacognitive, problem-solving, and social abilities. Through a National Science Foundation (NSF) grant, he developed a K–6 elementary geometry curriculum, *Logo Geometry* (published by Silver Burdett, & Ginn, 1991). With colleagues, he is working on the previously mentioned NSF research grant and is finishing a

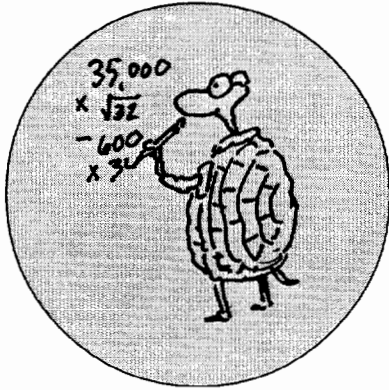
second NSF-funded project, *Investigations in Number, Data, and Space*, to develop a full K–5 mathematics curriculum featuring Logo. With Sarama, he is co-authoring new versions Logo for learning elementary mathematics. One, *Turtle Math*, is currently available from LCSi. Sarama and Clements are co-PI’s on the aforementioned *Building Blocks* project, which is developing mathematics software for preschool to grade 2 children.

Julie Sarama, Ph.D., is an assistant professor at Wayne State University, where she teaches mathematics content courses for pre-service teachers and research courses for graduate mathematics education students. She has studied teachers’ use of computer innovations and students development of mathematical constructs while working in computer microworlds. She is co-author of several *Investigations* units and of *Turtle Math* and has designed and programming new versions of Logo and other computer microworlds. She is co-principal investigator on the new *Building Blocks* project.

¹ Both agreed.

Douglas H. Clements
SUNY at Buffalo
Dept. of Learning and Instruction
593 Baldy Hall
Buffalo, NY 14260
Clements@acsu.buffalo.edu

Julie Sarama
Wayne State University
Teacher Education Division
Detroit, MI 48202
Sarama@coe.wayne.edu



Drawing Polyhedron Patterns

by WILLIAM J. SPEZESKI

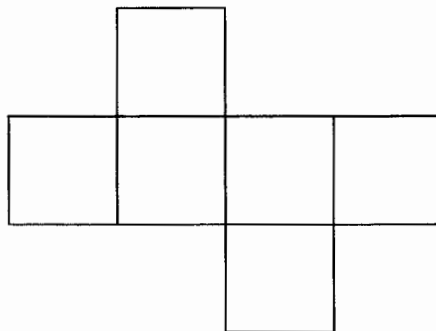
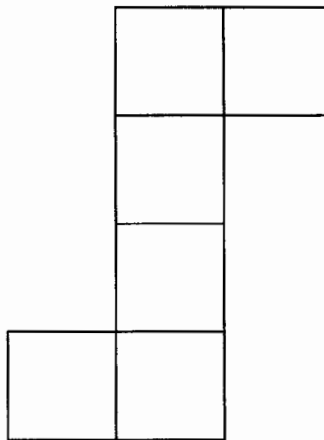
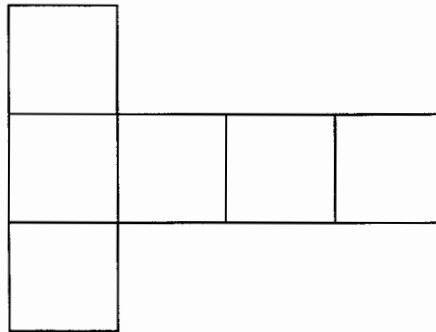
Drawing regular polygons with Logo is accomplished very easily by using a procedure like

```
to poly :n :l
  repeat :n [ fd :l rt 360/:n ]
end
```

where the parameters n and l are used to control the number of sides and the side length, respectively. The ever-present challenge for a teacher is to find innovative ways to weave the topic of polygons into an interesting project. One thought is to use the polygon procedure to create the faces of three-dimensional polyhedrons by drawing corresponding two-dimensional polyhedron patterns. Polyhedrons are simply three-dimensional objects whose faces are regular polygons. Cubes and pyramids are examples of polyhedrons. Once the patterns are constructed using Logo, they can be printed, cut, folded and then assembled into the corresponding polyhedron for a visual finale to the process. Let's consider two examples: the pattern for a cube and the pattern for an octahedron.

Constructing a Cube

To begin the construction of any polyhedron, we need a two-dimensional pattern layout to use as a guide. Figuring out the pattern layout can be part of the problem-solving process and the fun. In the case of a cube, there are several layouts, three of which are shown below:



We will draw the pattern on the top. We begin by examining the pattern for any repetition that occurs because that will make our programming job easier. We note that there are two rows of (three) squares, one vertical and one horizontal. A good programming strategy then is to have one procedure draw one row of squares and a second superprocedure that uses the row of squares to draw the pattern. The two procedures that create the pattern are

```
to row
  repeat 3 [ poly 4 50 fd 50 ]
end
```

and

```
to pattern
  seth 0
  row
  bk 50 rt 90 fd 50
  row
end
```

Executing the superprocedure PAT-TERN creates the pattern for the cube, but we will have a hard time assembling the printed pattern into a cube unless we create some tabs that can be glued to the sides of the cube to help hold it together. Here is a procedure that will draw a tab.

```
to tab :l
  fd :l lt 120
  fd 20 lt 60
  fd :l - 20 lt 60
  fd 20 lt 120
end
```


The tab is simply a trapezoid whose largest side length `:l` can be selected to match the side length of the polygon which forms a polyhedron face. The tabs can be added to the pattern by letting the turtle travel along the perimeter of the pattern and deposit a tab as needed. The tabs can be arranged in a multitude of ways. In fact, arranging the tabs requires its own bit of problem-solving. Another challenge, by the way, is to make the pattern fill the screen so that it will be as large as possible. It is difficult to assemble a small pattern. A final cube pattern can be drawn with

```
to cube
  pattern
    rt 90 tab 50
    fd 50 rt 90 fd 150 lt 90
    repeat 3 [ tab 50 fd 50 rt 90 ]
    lt 90 fd 50
    repeat 3 [ tab 50 fd 50 rt 90 ]
  end
end
```

The two-dimensional pattern for the cube, including the tabs, is shown next.

This pattern can now be printed, cut along the perimeter lines, folded along the interior lines, and finally assembled into a cube with the help of a little glue. To make the final assembly easier, it might be a good idea to glue the pattern onto a slightly thicker piece of paper before cutting it out.

The discussion so far has been about constructing a simple cube. For a more complex polyhedrons, there is a fair amount of geometry to consider as well as planning, particularly in the case of tab placement. But that is part of the challenge and the fun.

There are at least two types enhancements that can be made to make the polyhedrons more interesting. We can make structural changes to the face of the cube; we can also make structural changes to the edges of the cube.

Modifying the Polyhedron Faces

Patterns can be cut into the polyhedron faces to create an even more interesting visual effect with both the pattern and the assembled polyhedron. Two face

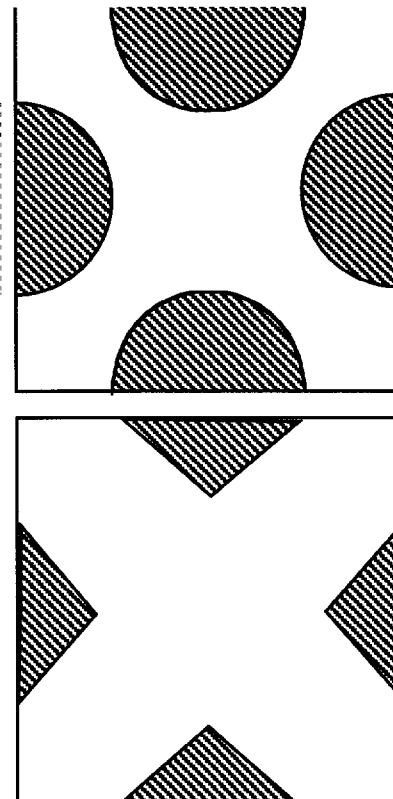
patterns for a cube are shown below.

If the cross-hatched area of the these cube faces are cut out, the resulting cubes will have a lattice-like appearance. Cutting the cross-hatched areas out is not as easy as cutting along the perimeter of the pattern. A hobby knife makes the job a little easier. The above examples are just two of the many possibilities for changing the appearance of the cube's face. The faces of other polyhedrons with different polygon shapes can be modified in a similar manner.

There is a special case of face modification that actually results in edge modification.

Modifying the Polyhedron Edges

If the areas to be cut out from the face indeed touch the edges of the face, then a "scalloped" edge is produced on the assembled polyhedron. Two examples of face cuts that create this effect for a cube are shown next.



In both the case of modifying a polyhedron face and/or modifying a polyhedron edge, the tabs may have to be adjusted slightly so that they don't overlap the cut-out area. This adjust-

ment is probably best left for the assembly stage of the project.

Constructing a Decahedron

The techniques used in creating a cube can be generalized to apply to all two-dimensional polyhedron patterns. The two procedures shown below combine with procedures POLY and TAB above to create the pattern for a decahedron, a polyhedron with 10 faces; each face is an equilateral triangle.

```
to face
  ; draw one face
  poly 3 100
  rt 30 pu fd 20 lt 30 pd
  poly 3 65
  rt 30 pu fd 20 setpattern
  5 pd fill pu bk 20 lt 30
  rt 30 pu bk 20 lt 30 pd
end

to decahedron
  cs fs ht
  ; draw the pattern
  pu setxy [-100 0] pd seth 180
  repeat 5 [face rt 60]
  fd 100 lt 60 fd 100 rt 180
  repeat 5 [face rt 60]
  ; add the tabs
  rt 60 tab 100
  pu fd 100
  repeat 4 [ fd 100 rt 60 pd
  tab 100 pu ]
  fd 100 rt 60 fd 100 rt 120
  pd tab 100
end
```

The pattern that results from executing procedure DECAHEDRON is shown below.

There are a few observations that are worth pointing out: The basic pattern consists of two figures, each a hexagonal shape with one missing sector. The corresponding Logo commands reflect that this hexagonal shape is repeated twice. Each face of the decahedron has been modified to create a lattice-like appearance. The areas to be cut out have been FILLED with a cross-hatching pattern. Additionally, the tabs have been FILLED as well. This helps to visually identify them. The commands to accomplish the FILLING do

not appear in the procedures above, but they can easily be added.

Some Polyhedron Patterns

To facilitate drawing some of the many polyhedron patterns with Logo, a sampling of polyhedron (tabless) layouts is shown below. Don't forget the tabs!

The patterns shown above are by no means unique. As noted earlier in the case of the cube, many patterns can produce the same polyhedron.

Conclusion

Creating two-dimensional polyhedron patterns is a very interesting Logo project. It combines drawing polygons with applying geometric perception in both two and three dimensions. The purpose of this article is to be a catalyst for creativity, but not be complete in its coverage of polyhedrons. There are many books on the subject, though. I've mentioned the names of two books that I found particularly insightful, at the end of this article.

The process of creating polyhedron patterns can be extended to cutting, folding and pasting in the final assembly. In addition, multiple polyhedrons, usually of the same kind and size can be glued together to produce some sensational three-dimensional effects. But note that now we have crossed over from the world of Logo and wandered into the world of three-dimensional design and sculpture. Enjoy it. You may even get lost there!



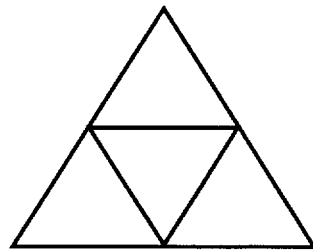
References

Chatani, Masahiro, and Nakazawa, Keiko (1994). *Pop-up Geometric Origami*. Tokyo, Ondorisha Publishers, LTD.
 Wong, Wucius, and Reinhold, Van Nostrand (1977). *Principles of Three-Dimensional Design*. New York.

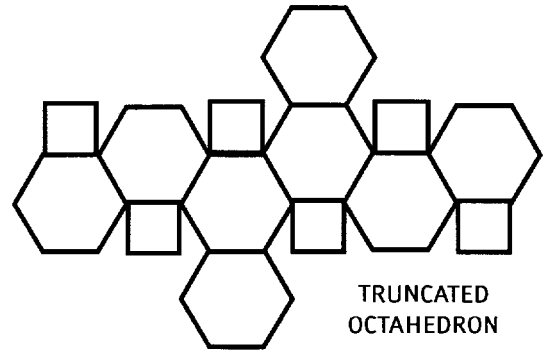
About the Author

Bill Spezeski teaches at Massachusetts College of Liberal Arts and is a long-time contributor to Logo Exchange. Look for other articles by him in future issues.

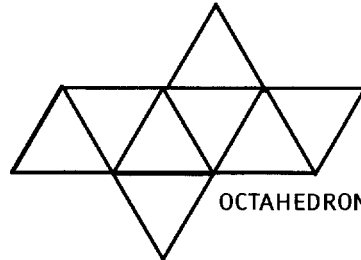
Bill Spezeski
wspezski@mcla.mass.edu



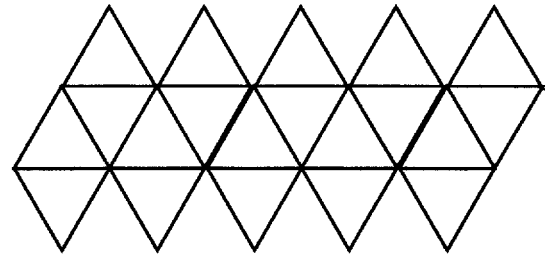
TETRAHEDRON



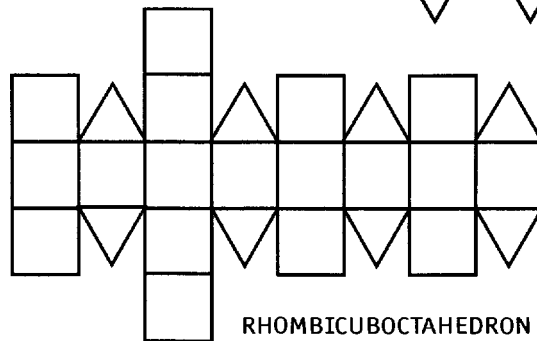
TRUNCATED OCTAHEDRON



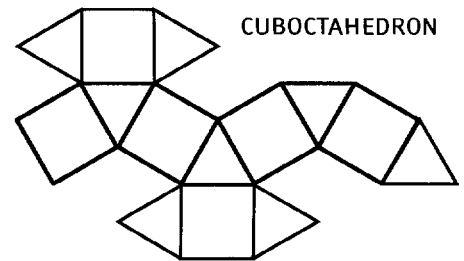
OCTAHEDRON



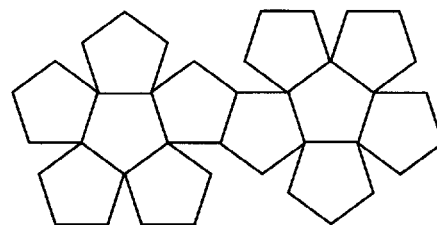
ICOSAHEDRON



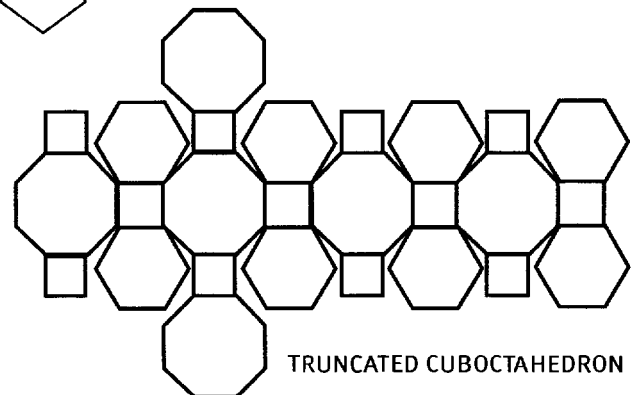
RHOMBICUBOCTAHEDRON



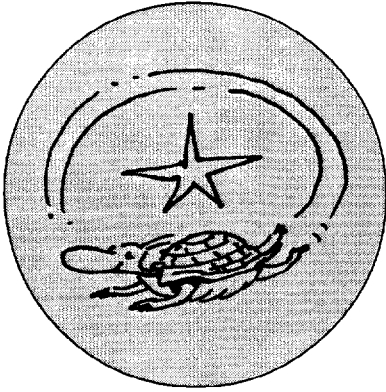
CUBOCTAHEDRON



DODECAHEDRON



TRUNCATED CUBOCTAHEDRON



The Chase: Part 2

Last time I sketched out the rudiments of a predator-prey model in which lynx chase rabbits but also depend on the rabbits for survival. In this column I will augment this model with a simple graphing facility.

With any dynamic system, data collection is an important analysis component. The collection and storage of data points allows for later review and study, and plotting the data points offers a compact way to represent and summarize this data.

StarLogo as a graphical simulation system supports "real-time" activity monitoring in the Graphics window. Additionally the Plot window supports two dimensional graphs to be constructed as the simulation is running, with six variables represented. Although each axis can utilize a different scale, the most usual Y scale is time, as measured in StarLogo time ticks.

I will show how to add graphing to the lynx and rabbits model, and add some other features to make this model more useable. Then I will discuss some investigations for further work with this model. It is important to build the model from the last column since the code here is based on, and added to that model.

In order to use the graphing feature, a setup procedure is used to establish the plot pen colors, the axis ranges and the text for the X and Y axes. (In these procedures, I illustrate an alternative comment method by inserting comments directly in the procedures.)

```

to setup-plot
  ;; start with blank graph
  clearplot
  ;; select a pen, reset it to its origin,
  ;; and set colors to match turtle colors
  pp1 ppreset setppc dk-orange
  pp2 ppreset setppc lt-pink
  ;; label the axes
  setplot-ylabel "Pop.
  setplot-xlabel "Time
  ;; set range of values for each access.
  ;; Y-axis starts at twice total
  ;; population of lynx and hare
  setplot-xrange 0 100
  setplot-yrange 0
  (2 * (lynx-initial + hare-initial))
end
    
```

The setup procedure must be augmented to call setup-plot:

```

to setup
  ca crt lynx-initial + hare-initial
  setpc white
  setup-all
  setup-plot
end
    
```

In order to cause points to be plotted, the graph-point procedure must be added, and called in the move procedure:

```
to move
step
ask-lynx [lynx-chase-hare
          lynx-death]
ask-hare [hare-reproduce
          hare-death]
graph-point
remove-dead
end
```

The graph-point procedure relies on a slider called graph-on? that lets the user turn plotting on and off during the simulation run. Create a slider with a range of 0 to 1. It also works best with two observer procedures that return the current count of each breed:

```
to total-lynx
output color-count brown
end
```

```
to total-hare
output color-count pink
end
```

These are useful also in monitor windows so the user can keep track of the populations. Create two monitors, one for total-lynx, the other for total-hare.

Now for the graph-point procedure. Each time it is called, it checks the graph-on? slider, plots one point each for the current total hare and lynx counts, and then resizes the graph if the plot pen has run off the page.

```
to graph-point
if graph-on? = 1
;; select a pen and plot one point
;; for lynx and one for hare
[pp1 plot total-lynx
 pp2 plot total-hare
;; check if each axis max. is reached;
;; if so, extend the plot max
if ppxcor > (plot-xmax - 1)
[setplot-xrange 0 int(ppxcor * 1.5) ]
if ppycor > (plot-ymax - 1)
[setplot-yrange 0
 (min 300 int(ppycor * 1.5)) ]]
end
```

Before running the simulation, change the Settings in the Edit menu so that the screen dimensions are 25 x 51. This will permit a smaller simulation space with characters that look a little like animals. Feel free to change their shapes if you want, being aware that setcolor doesn't affect shapes other than turtles.

As the simulation is running, open and look at the Plot Window to see how the hare and lynx populations change over time. The hare population goes through many oscillations which form a long term trend. The total number of lynx rises and falls, somewhat out of phase with the hare population. How can you explain the short and long term trends? Think about what in the program, and in real life, causes the populations to get larger and smaller.

After a long time, you might want to save and print your graph for later analysis and comparison. Stop the simulation by clicking the go and check-vacuum buttons again. Capture the Graph using the menu File/Export/Plot Window. This will create a Pict file on your hard disk which you can print using SimpleText. Label this graph "First Run".

After running this model, you may find that one or the other species becomes extinct. Since these two species are dependent on each other, the loss of all of one species certainly affects the other, but completely differently. Can you predict how?

Some additional procedures will simulate the immigration of new members of a species when there are none left in the space. A button called check-vacuum that runs forever will effect this with the following procedures:

```
to check-vacuum
;; do this every 6 seconds
every 6
;; if there are no hare,
;; allow immigration
[ifelse total-hare < 1
 [migrate-hare]
;; otherwise check on the lynx
 [check-no-lynx]]
end
```

```
to check-no-lynx
;; check if there are no more lynx;
;; 30% of the time, bring another in
if total-lynx < 1
[if random 10 < 3
 [migrate-lynx]]
end
```

```

to migrate-hare
;; create a new hare
crt 1
;; new turtle will not have a
;; breed set yet, and it won't
;; be a lynx
if not (breed = lynx)
  [genetics 0]
end

```

```

to migrate-lynx
;; create a new lynx
crt 1
;; new turtle will not have a
;; breed set yet, and it won't
;; be a hare
if not (breed = hare)
  [genetics 1]
end

```

Be sure to create the check-vacuum button with 'forever' checked. By pressing this button during a simulation, whenever the lynx or hare become extinct, eventually a new animal will wander into the area.

An interesting side effect of the way this model has been programmed is the way the lynx seem to get confused when they are close to one another, and fail to chase the hare until they are farther apart. Spend some time watching the lynx behavior, especially when there are few animals on the screen. How could this behavior be explained in real life? Can you think of a way to change the model to avoid this trait?

What is needed is a change to the lynx-chase-hare procedure so that the lynx will repel each other (and therefore concentrate more on catching hare). Predict what you think will happen as a result. Why?

[Hint:] To make this change, look carefully at the lynx-chase-hare procedure. In it is the line:

```
if (breed-of closest) = hare
```

followed by the code that lynx use when they are close to a hare. What is needed is an else part which deals with what to do when the lynx is close to another lynx. You can do this by adding a call to a new procedure called lynx-avoid-lynx. Write this new procedure with code that turns the lynx 180 degrees AWAY from 'closest' (the other lynx). Copy, paste and modify any code that looks like it might be similar to what you want to do and fix it to work for repelling lynx.

What effect does this change have on the predator/prey relationship? Print and label a graph of a simulation run with this change.

The lynx seem to have very long range vision—they can spot hare clear across the entire area. We can give the lynx nearsightedness to limit their ability to see hare far away. How might this change the situation?

[Hint:] Look in the eat-hare procedure and add another test that checks whether dist-to-closest is less than some amount (you decide) before setting its heading toward the closest hare. Try different values if what you select doesn't make any difference in the model, or add a slider that allows you to change the vision while the simulation is running.

What effect does this change have on the predator/prey relationship? Print and label a graph of a simulation run with this change. Compare this graph with one from the earlier runs. What are the differences? Is this a more realistic model?

The program we have written defines a set of rules for the hare and lynx to follow in their artificial lives. Can you write these rules in plain English? What will be the effect of altering these rules? For example, what will happen if the life span of the lynx is reduced?

Predator-prey models are rich in explorations. Feel free to write your own.



About the author

Alan Epstein is the Director of Technology for Watertown Public Schools in Massachusetts. He has also spent 24 years developing software, most recently commercial modeling and simulations systems. He holds an M.Ed from the Harvard Graduate School of Education's Technology in Education Program.

Alan Epstein
Alan_Epstein@watertown.k12.ma.us



Logosium '98 Report

by LOGO EXCHANGE STAFF

The sun is shining and the surf is up, so why would a busload of technology educators climb aboard a bus at 8 a.m. on a beautiful Sunday morning in San Diego and travel all the way to Pepperdine University's Irvine campus? Call it the lure of Logo. Logosium '98 was an action- and thought-packed day that combined reunions with Logo friends from the past, meetings with new-to-Logo adventurers, and opportunities for hands-on sessions with folks who have been dedicated Logo users for many years. Educators from across the



Brian Silverman

United States, Brazil, Mexico, Australia, and New Zealand participated.

Gary Stager arranged a day that was long on Logo and packed with information. Two morning breakout sessions allowed Logosium participants to choose topics that suited their interests and level of Logo expertise. The program offerings included:

- Getting Started with MicroWorlds—Fran Reedy
- Getting Started with UCB Logo—Brian Harvey

- Introducing the Programmable Brick—Frank Mead
- Logo on the Web—Ron Revere
- Intro to Robotics with Control Lab and LEGO CAD—Frank Mead
- Learning through Game Design—Hope Chafian
- Mathematical Recreations and Virtual Computer Archaeology—Brian Silverman
- Integrating LEGO Logo into the Math and Science Curriculum—Sandra Turner



- Drawing Polyhedron Patterns — William Spezeski
- Creating Interactive MicroWorlds Projects—Gary Stager
- Computer Science Logo Style—Brian Harvey
- Learning with Starlogo—Brian Silverman

Brian Silverman's presentation of Starlogo was a highlight for many. The beauty and elegance of the display and the realization that this was a powerful tool for expressing mathematical ideas were immediate and sparked a lively discussion of how Starlogo could be used in the classroom.

Brian Harvey's presentation also brought even the most jaded Logo teachers to a new degree of understanding of Logo's power as an educational tool. And the good news is that many of this year's participants were new to Logo, excited by its potential for getting at powerful mathematical ideas and stimulated by the devotion of old-time Logo users to Logo as a tool for enriching students' mastery over technology, mathematics, art, and space. Many commented that Logo is a major vehicle for attacking the "girls and spatial reasoning" situation, so the acceptance of Logo as a major tool for intellectual development was a welcome outcome of the day.

The masterful arrangements included a box lunch for everyone and a



Fran Reedy



Gail Marshall

sunlight patio where followup discussions of the morning's topics could be conducted over veggie sandwiches and ice-cold sodas. It was an opportunity for sun-starved Easterners to catch a few rays of sunlight, but it was also a time for forging Logo friendships, debating the merits of one or another approach to using Logo, and a time to catch up on classroom applications of Logo stories.

In his after-lunch keynote, Jeff Richardson urged the Logosium participants to reconsider the term "surf the Web." Surfing, said Australian veteran surfer Richardson, was not a casual sport. Surfing called for planning, hard work, tired days, and stamina. So surfing the Web should not be a matter of a few casual mouse clicks, but it should be a time of preparation, planning, mental labor, and intellectual discipline.

In closing remarks, Gail Marshall urged attendees to consider the political climate in which Logo and other constructivist activities are conducted. Today's stakeholders—parents, community members, media, students, teachers, and administrators—are working in an era where accountability governs most, if not all, educational decision making. So Marshall urged participants to think carefully about ensuring that they were aware of outcomes—how their were students learning and how well they were learning

(and not just in terms of standardized tests but in the broader areas of building hypotheses, testing hypotheses, looking for alternatives, and seeking patterns of thought that work in many different situations and other manifestations of intellectual growth). Marshall also urged teachers of Logo to tell their stories. How easy is it to work in a Logo-saturated environment? What problems do they encounter, and how do they solve them? How do they know when they're making a difference, and how do they communicate the importance of what students learn?

A spirited discussion followed Marshall's comments with the general consensus clearly in favor of telling the Logo story more often to a wider audience in ways the audience will understand and value. Camaraderie spilled over into the bus back to San Diego with a welcome stop for clams, crabs, lobster rolls, lemonade, sodas, and beer at Joe's Crabshack in Newport Beach.

The goodwill, curiosity, commitment, and interest that flowed through the day made it clear that Logosium fits a need in the educational technology community—a place to get hands-on time with Logo by working with peers and pros, a chance to debate the relative merits of one or another way of working with Logo and an opportunity to recharge intellectual batteries by listening to people dedicated to Logo as an educational tool and philosophy of learning.



Hope Chafian

Logosium '98 Scrapbook



Jeff, Brian, and Dave



Bus trip



Jeff and Brian



Waiting for dinner



New Year's Resolutions

by **SUSAN OHANIAN**

One of the true glories of being a teacher is that life really does begin anew each September (other months outside of North America). What happened last year is relegated to folklore, and we get to try again. Each September, students and teachers alike get the chance to start again. In this spirit of beginning again, I have a few proposals for new year's resolutions.

Take time to ignore the skills

Increasingly, teachers and students alike are surrounded by imperatives to pile up the skills. Everybody from TV pundits to newspaper op-ed writers to Congresspersons to corporate CEOs preach Standards, and in most peoples' minds Standards are translated into memorizing the times tables in third grade and quadratic equations by eighth. There is considerable pressure to move term paper writing lower and lower. I know of schools where parents are asking for research papers in second grade.

Beset as we are by agendas and products promising us curricula and machinery to teach twice as much in half the time, let's resolve instead to stand firm for attempting half as much in twice the time. One way to start on this road is to realize there is no such thing as a basic skill. What we call skills are merely disjointed items that appear on standardized tests. They have no aes-

thetic value, are of no practical use to anybody, and should not be allowed to invade our classrooms.

Make 1998-99 the year without answers

Let's restrain our pedagogical greed for answers, instead making 1998-99 the year for questions. Around the turn of the last century, Scottish novelist and poet George MacDonald complained, "I offer you roses and you boil them." Now, one hundred years later, what would happen if we made 1998-99 the year of rejoicing in roses rather than boiling them? What if we made this a year of enjoyment, not interrogation?

Avoid snake oil

We're hearing a lot these days about "skills for the millennium." They don't exist. The illicit progeny of the snake oil marketed in the 19th century as universal cure-alls for every thing from sexual dysfunction to irregularity to depression, promises made in the name of millennium skills are just as phony. We must not confuse racket with reality. Even if we did have some crystal ball for seeing future job possibilities for our students, we should resist these futurist lures. A teacher's job is not to teach for the future. As teachers, always, always, we must concern ourselves with the present lives of the children in our care.

Don't mistake convenience for progress

It is no surprise that teachers, whose professional lives are filled with so much inconvenience, would sometimes get enticed into mistaking convenience for progress. The truth of the matter is that there isn't one school in 50 where the institution of the photocopy machine has brought about an improvement in the educational experience offered children. In most schools, the copy machine just makes it easier to hand out lots of killing skill drill papers.

Discontent with the status quo, not mechanical wizardry, is the first step toward progress. This is why, so far, computers have failed to revolutionize education. Most software purchased by most schools, whether it's teaching cblend or the 38 rules for capitalization, is just whistling in the pedagogical dark.

Celebrating individual idiosyncrasy

Educational camp followers—whether they're publishers or methods entrepreneurs or bureaucrats or politicians—are always on the lookout for the product or system or technique that can bypass the vagaries of individual teachers and students and thereby truly standardize education. If computers are employed to deliver such a stan-

See RESOLUTIONS (Page 15)

Logo Online

A growing collection of Logo materials is available on the Internet. From time to time, Logo Exchange will point you in their direction. Below is a starter list for you to explore. Many of these sites contain links to numerous other Logo-related sites.

Share a sample issue of Logo Exchange with friends at

<http://moon.pepperdine.edu/~gstager/logoexchange/>

Browse through books recommended or mentioned in *Logo Exchange* at

<http://moon.pepperdine.edu/~gstager/logoexchange/logoexchangebooks.html>

Michael Tempel and Hope Chafiiian have created VideoGameWorks, a collection of kid-created MicroWorlds games playable on the Web. Point your browser to

<http://el.www.media.mit.edu/logo-foundation/VGW/>

Get the free MicroWorlds Web plugin and explore a library of MicroWorlds projects at

www.microworlds.com

Seymour Papert recently delivered a terrific speech in England. Read, listen, or watch it at

www.connectedfamily.com/frame4/cf0413seymour/recent_essays/cf0413_cherry_1.html

Despite its conspicuous lack of Logo, LEGO's Mindstorms (programmable RCX brick) is making waves on-line. LEGO's home Mindstorms site is

www.legomindstorms.com

and the educational division's site may be found at

www.lego.com/dacta/robofab/defaultjava.htm

The following sites are either about the growing programmable brick hacker community or contain resources for programmable brick users.

- www.salonmagazine.com/21st/log/
- www.wired.com/news/news/culture/story/15171.html
- www.crynwr.com/lego-robotics/
- <http://graphics.stanford.edu/~kekoa/rcx/>

Want to know where the programmable brick research is heading? Check out the Cricket Web site at

<http://fredm.www.media.mit.edu/people/fredm/projects/cricket/>

and a paper, "Programmable Bricks—Toys to Think With," at

www.almaden.ibm.com/journal/sj/mit/sectionc/martin.html

The massively parallel and wonderful StarLogo 2.0.4 for the PowerMac and 2.0.2 for 680X Macs is available for download at

www.media.mit.edu/~starlogo/download

Check out the StarLogo Web site at www.media.mit.edu/~starlogo.

You will find updated documentation, more sample projects, user community information, pointers to related work, and a comprehensive list of all new features. To join a list of StarLogo users, point your browser to

www.media.mit.edu/~starlogo/community/

Other StarLogo sites are full of interesting challenges and resources. They include:

Penn State StarLogo Sites at

www.psu.edu/dept/beeive/starlogo/starlogo.htm

Tufts University Connected Mathematics & (mathematically enhanced) StarLogo T1.5 Software at

www.ccl.tufts.edu/cm/

Maine StarLogo Communities (home of the PC beta-version of StarLogo) at www.asap.um.maine.edu/starlogo/

Brian Silverman and Mitchel Resnick coauthored a dynamic essay on related issues called "Exploring Emergence." Find it at

<http://el.www.media.mit.edu/groups/el/projects/emergence/>

Terrapin Logo and PC Logo product information and other resources are available at

www.terrapinlogo.com

Windows users may be interested in MSWLogo 5.3, which is available for free download from

www.softronix.com

Brian Harvey's Web page contains many papers written by Brian, information on his books, and free downloadable versions of Berkeley Logo. Go to

<http://http.cs.berkeley.edu/~bh/index.html>

Berkeley Logo is available for Macintosh, Windows, and Unix computers.

Gary Stager's Web site, www.stager.org/logo.html, has a large collection of resources and links of interest to Logo-using educators and learners.


The Logo Foundation and the Global Schoolnet Foundation sponsor the Logo-L listserv. Subscribe to this listserv and join a global community of Logo users via e-mail. No fancy browsers or high-speed data lines are required.

To subscribe, send an e-mail message to majordomo@gsn.org and put the words subscribe logo-l in the body of the message. All of the postings to the listserv are archived on the Web as well at

<http://archives.gsn.org/logo-l>

Speaking of the Logo Foundation, its Web site at

<http://el.www.media.mit.edu/groups/logo-foundation/index.html>

has links to numerous other Logo-related Web sites and a catalog of Logo materials available for purchase. 

ISTE BRINGS THE WORLD OF TECHNOLOGY CLOSER TO YOU.

By drawing from the resources of committed professionals worldwide, ISTE provides support that helps educators like you prepare for the future of education.

As an ISTE member, you benefit from a wide variety of publications, national policy leadership, and our work with Teacher Accreditation.

You also enjoy exciting conferences, global peer networking, and graduate-level Distance Education courses.

So if you're interested in the education of tomorrow, call us today.



iste International Society for
Technology in Education

Teachers Helping Teachers Use Technology in the Classroom

WE'LL PUT YOU IN TOUCH WITH THE WORLD.



iste

International Society for Technology in Education
Administrative Office
1787 Agate Street, Eugene, OR 97403-1923 USA
Phone: 541/346-4414 Fax: 541/346-5890
Internet: iste@oregon.uoregon.edu

Non-Profit
Organization
US Postage
PAID
ISTE