Winter 1999

Volume 17 / Number 2

Logo Exchange

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

PERSPECTIVES ON PAPERT

INSIDE

iste

Logo Exchange

Volume 17 / Number 2

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, *Curriculum Administrator* Magazine
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 Chafiian, Secretary/Treasurer
Gary S. Stager, Editor

SIG Coordinator
Tom Magness

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*
M. D. Roblyer *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. 4 .... April 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.

© 1999 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.

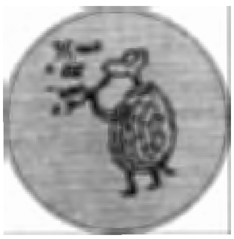
POSTMASTER: Send address changes to *Logo Exchange,* ISTE, 480 Charnelton St., Eugene, OR 97401-2626 USA. Periodicals postage paid at Eugene, OR. USPS# 000-554. ISTE is a nonprofit organization with its main offices housed at the University of Oregon. ISSN# 0888-6970.

This publication was produced using Aldus PageMaker®.

Logo Exchange

Winter 1999 Vol. 17 / No. 2

Contents



ARTICLES

COLUMNS

iste

EDITORIAL / GARY S. STAGER

Never Satisfied, Only Gratified

Those were the frequently uttered words of my old trumpet teacher, William Fielder. You may enjoy the momentary pleasure of success, but persistence and hard work are required if you wish to achieve greatness. This simple poetic phrase reminds me of Dr. Seymour Papert.

Papert's life, work, and ideas have inspired countless people around the world and acquainted millions of children with the joy of learning. While reasonable people may differ about whether Papert is the Father of Logo, he can surely be considered its loving mentor. In fact, Papert was one of the first people to suggest (more than 35 years ago) that computers could play an important role in learning. Thirty years ago, Papert, Alan Kay, and Cynthia Solomon were predicting that every child would own a portable computer. Papert's previous accomplishments in mathematics, cognitive psychology, and artificial intelligence led credibility to such predictions.

In Papert you find the rare futurist. He not only launched the idea of learner-centered computing, but then spent several decades expanding his theories while actually building things (both software and ideas) used by others. His work shares more with Thomas Edison than most "ivory tower academics."

Logo offers learners a powerful intellectual laboratory and vehicle for self-expression while providing teachers with a catalyst for rethinking the nature of teaching and learning. Logo is an object to think with, both for the learner and for the people who are thinking about the thinking of the learner. Seymour Papert has provided countless educators with the thrill associated with students performing intellectual feats they never before believed to be possible. He invited educators into a community of powerful ideas by giving voice to their experiences and encouraging the sharing of learning stories. We have been given a way in which to discuss profound ideas without being overburdened by ostentatious vocabulary or overly technical theories. Best of all, we are encouraged to interpret these ideas in personally resonant ways.

Papert's books about education—*Mindstorms, The Children's Machine* and *The Connected Family*—take the reader through the development of his thinking about learning with computers. We are challenged to question our own assumptions in order to take steps toward enriching the learning experience for children. (Carolyn Dowling explores these books later in this issue.)

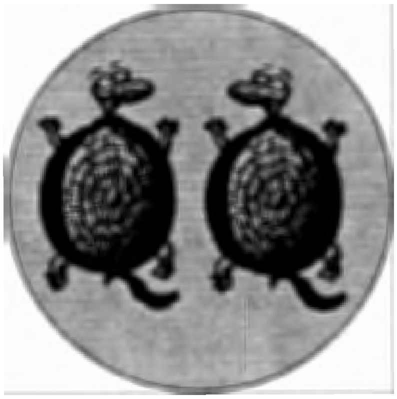
It has always pleased me that Papert's books have enjoyed no serious criticism in academic circles. People may disagree with a point or two in the books, but there has been no serious piece of scholarship arguing against the ideas found in *Mindstorms*. This does not, of course, mean that Logo, or even Papert personally, has been free of criticism.

The attacks on Logo, with few exceptions, have not been fought on the battlefield of ideas but rather in the marketplace. Logo is bad for business.

If kids construct their knowledge and express themselves in an environment designed to have "no threshold and no ceiling," then you are not likely to buy lots of other software products. Schools not disposing of old computers because they are just perfect LEGO TC logo workstations don't run out and buy as many new computers each year. While the Software Publishers Association may honor Papert with a lifetime achievement award, its member companies conspire to keep Logo-related presentations off far too many educational technology conference programs. Logo is also bad for the business of schooling because it encourages concerned adults to rethink the nature of teaching and learning.

The most noxious attacks on Logo are acts of omission. As a university teacher educator, I receive countless textbooks on the theory, history, and practice of educational computing for my consideration. The majority of these texts don't disagree with Logo research or the theories of Seymour Papert. They don't mention them at all. Most of these books purporting to provide an intellectual and/or historical understanding of educational computing ignore four decades' worth of research and classroom practice. This is unacceptable and intellectually dishonest. One rare exception is *Designing Multimedia Environ-*

See NOT SATISFIED (Page 4)



Turtle and the Moon

It started out as an ordinary event. I took my son Kyser outside one evening to observe the moon in its first quarter. Before long, however, it turned into an extraordinary event and then (a bit later) into an amazing event! And finally, it gave me another perspective about what we can learn from the turtle.

But I'm getting way ahead of myself! Let me back up and tell you what happened.

When we went outside, it was on the downside of dusk and the sky was just darkening. The moon was spectacular. We could see the brilliantly illuminated right side of the moon and the uneven shadows along the terminator between light and darkness. We also noticed that Jupiter was shining brightly above and to the left of the moon. It was simply beautiful.

About three hours later, I had to go outside for something. While out, I happened to take a look at the moon. What I saw sent me running back into the house to get Kyser!

From their earlier position, the moon and Jupiter had moved across the sky toward the west. But Jupiter was no longer to the left of the moon; it was directly above it, lined up with the terminator.

We discussed what we were seeing, and concluded that the westerly movement of the moon-Jupiter system was due to the rotation of the earth. Just like the sun, it was moving from east to west. But we could also see the result of the revolution of the moon around the earth as it moved apparently eastward beneath Jupiter!

We decided to come out again in three more hours, and discussed what it might look like then. Sure enough! We saw that the moon and Jupiter had moved further west across the sky. But the moon had also moved eastward with respect to Jupiter. The shining planet now appeared above and to the right of the moon. Relative motion in action!

Such extraordinary events are wonderful opportunities to observe and contemplate the rather complex movements of the heavenly bodies.

Now, here's the amazing part. During the next afternoon, I logged onto a live camera site on the Internet (**www.africam.com**) that was situated by a waterhole in southern Africa. I wanted to see what animals might be drinking there at dusk.

Instead of seeing animals, however, I was treated to a beautiful view of the moon! The camera host had recognized the spectacular sky scene and had trained the cam's lens on it for others to enjoy as well. There on my computer screen was Jupiter, but it was below and to the left of the moon! And the moon was lighted on the left side instead of the right! Amazing!

Kyser and I enjoyed this series of observations very much. But it also reminded me of the Logo turtle. All of us have seen the fascinating situation of a child trying to figure out how to turn a turtle that is facing towards the bottom of the screen. Whose "left" is right, that of the child or that of the turtle?

Intellectually, I knew that the same side of the moon was illuminated each night. But I had to do some real mental gymnastics to picture why the left side appears to be illuminated for observers in the Southern Hemisphere. Like many children wondering about the upside-down turtle, I found myself contorting my body, turning sort of upside down to look at the moon and Jupiter, so I could better imagine what it must look like for someone in southern Africa. Finally, I could make comfortable sense of it.

This ordinary, extraordinary, and amazing series of events reminded me once again that some turtle aspects are challenging for young minds; yet, they are accessible. Ideas such as relative motion and relative direction sound complex, but they are all around us. The rich learning environment of the Logo turtle provides the stimulus to examine these ideas in ways that are meaningful to us.

We could wish no better for our students. LX

FD 100!

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

BACK TO SCHOOL / Continued from Page 2

ments for Children by Allison Druin and Cynthia Solomon.

Ask a room full of educators who know something about Logo to brainstorm a list of the most frequently heard criticisms. The list always includes things such as: it requires teachers to learn new things; it requires too much class time to do something worthwhile; it's hard to assign a letter grade; or it doesn't fit neatly into traditional curriculum areas. These are not criticisms of Logo as much as they are criticisms of school. Logo is an embodiment of that criticism.

Seymour Papert possesses qualities that set him apart from many other great thinkers. He not only loves to learn, but he relishes other people's learning and helps us see the magic in our own learning. One of Seymour's prize pupils, Idit Harel, speaks in a glorious interview for next issue about Papert's playfulness. He loves toys, games and puzzles. His sense of humor is infectious and he is fueled by the hard-fun of children. Papert's humor and playfulness often lead to very important research.

Papert also cares a great deal about educational equity and believes that learning provides a vehicle for overcoming social injustices. His work has always been concerned with disadvantaged communities, whether they be in New York, the Roxbury section of Boston, or developing countries. Seymour's current projects include working in Thailand (see his Teacher Feature in this issue) with the Job Corps as well as a juvenile girls prison in Maine.

His willingness and ability to work with the corporate world has led to criticism from within the Logo community but has also produced actual products that benefit children. Papert does not view popular culture with disdain but as a variable necessary in any formula for understanding the way children learn now and in the future.

I have come to appreciate Seymour Papert's enormous contributions to the world of ideas in three ways.

1. His ability to generate brilliant theories.
2. His willingness to risk criticism for generalizing how those theories would look in practice.
3. His ability to predict how those ideas would be assimilated and misinterpreted by the institutions they challenge.

In many ways, Papert reconceptualizes Dewey, Montessori, Vygotsky, Piaget, A.S. Neil, and other progressive educators in a contemporary computer-rich world. His ideas are built on the shoulders of the great educators who came before. Moreover, Papert helps us see the tactical errors of our predecessors and the new opportunities that emerge with the widespread availability of personal computing devices. We are encouraged to use our imagination, to dream, to play.

My answer is that if you have a vision of Someday you can use this to guide what you do Monday. But if your vision of where it is going is doing the same old stuff a bit (or a lot) better, your efforts will be bypassed by history (Seymour Papert, 1998).

This issue contains a collection of perspectives on Papert from an international cast of contributors. Many of these contributors know Seymour Papert through only his work. I am grateful for their contributions and as always thank Peter Reynolds for his wonderful illustrations. Point to **http://www.stager.org/planetpapert.html** for an extensive collection of articles, papers and speeches by and about Seymour Papert.

Gary S. Stager

Gary

Gary Stager, Editor-in-Chief
logoexchange@stager.org

Jeff Richardson's Wild Ride

by JEFF RICHARDSON, INTERNATIONAL EDITOR

I rode with him
in taxi once
Only for a mile and a half
seemed like it took a couple of months
("Lenny Bruce" by Bob Dylan)

Seymour was a keynote speaker at the World Conference on Educational Computing in Sydney in 1990. I attended the conference as a humble delegate.

Our paths crossed in a downtown radio studio. Seymour was there to do a "serious interview." I was using a Tardis booth to make my regular weekly spot on Mel-bourne radio. We both finished up at the same time.

So I introduced myself and suggested we share a taxi back to the conference center. During the ride he generously listened to my crazy research ideas, interjecting ideas, and elaborations of his own. We were still talking animatedly when the cab pulled up at the conference center, where dignitaries were waiting to usher Seymour into some inner sanctum and his next appointment.

He stayed in the cab talking with me for a further quarter of an hour, the meter running, and frustrated suits on the footpath calling "Dr. Papert, we really MUST GO!" He made them crazy as he ignored them and continued our conversation about the usual things: children, computers, and powerful ideas.

TEACHER FEATURE

Mr. Condom

by SEYMOUR PAPERT

I recently came back from a trip to Thailand where I am working on a project to develop uses of digital technology for learning in so called "developing" countries. While there I met a remarkable educator who has provided me with what has become my favorite learning story.

He earned the title Mr. Condom (though when they are being polite he is more usually addressed as Mr. Mechai Viravaidya, Chairman of the Population and Community Development Association (PDA), by bringing a brilliant educational methodology the problem of encouraging villagers to practice birth and STD control. His procedure is this.

He goes to a village meeting, holds up a condom and asks if anyone knows what to do with it. The tension and silence is palpable. So he says: "Well, look." He unwraps the condom, puts it to his lips, and blows it up like a balloon. The tense silence is broken by a few giggles. So he continues in the same spirit . . . bouncing the balloon, blowing up another, trying to juggle . . . generally fooling around. When the crowd seems to entering the spirit of fun he hands out condoms to everyone urging them to join in the fun.

Then he goes away.

And if you look at the statistics you see that in the places where Mr. Condom has performed the birth rate has gone down.

Mr. Mechai explained to me that if he stood up there and spoke about sex, people would not listen. Besides, doing so would be irrelevant to the real problem. These villagers know about sex and would be perfectly capable of figuring out how to use a condom if they were not so uptight about it. What was needed was to make them have a more relaxed relationship with the thing.

Contrast his approach with what happens in many American sex education classes. Teacher draws on the board, or puts up a chart showing the plumbing of the human reproductive system. "Now this is how a condom works . . ."

The difference between Mr. Mechai and the approach of the American sex education class applies much more widely than to condoms. In fact I believe that Mr. Mechai has as much to tell us about math education as about sex education. Because if kids were not too uptight about fractions to play with them, they would find it easy to figure out how they work.

So let me tell you another story in which my then doctoral student Idit Harel did something for fractions very similar to what Mr. Mechai did for condoms.

It happened in the Hennigan School in Boston. Idit had given a fourth-grade class an assignment to make software that would teach something about fractions. The students had access to enough computers for each one to work on an individual product. They had enough time—a school period a day for most of a school year—to do a serious job. And they had enough support for learning the programming skills needed to carry out their projects using Logo (LogoWriter at the time) as their programming language.

All students were expected to deal with some aspect of fractions. But they could choose which. Most chose some aspect of schoolish knowledge about manipulating fractions. But some did something very different.

Debbie's software was very different indeed. She described the insight she wanted her software to convey as "There are fractions everywhere. You can put them on top of anything." To show this she would draw a picture and show that you can find many fractions in it. She spent a lot of time doing that and programming the computer to show the result.

Now I forgot to say that before this experience Debbie was a very poor math student and when interviewed about fractions showed not only a pitiful lack of knowledge but also an extreme reluctance to apply her mind to thinking about them. After her software writing experience she scored, for the first time in her life, at a superior level on standardized school math tests.

What happened? This goes against the grain of conventional wisdom in the school world. "If you want students to score well on tests about fractions, teach them the stuff they will be tested on." From this point of view Debbie was wasting her time with the software. Just like the villagers were "wast-

See MR. CONDOM (Page 12)

Dancing with Seymour

by TOM LOUGH, FOUNDING EDITOR

The dance is a powerful metaphor in many contexts. However, I was unprepared for how deeply such a metaphor might affect me when I first encountered Logo and Seymour Papert. Let me tell you how it was . . .

In the fall of 1981, I was a physics graduate student at the University of Virginia. By a series of fortunate circumstances, I was able to audit a new course offered by a couple of education professors named Glen Bull and Steve Tipps that focused on a new computer language called Logo.

The class used recently unpacked TI-99A computers with TI-LOGO cartridges. Glen and Steve taught in a highly interactive manner, allowing each of us to "play" with the turtle and to try out many different ideas.

The assigned text for the course was a paperback book called *Mindstorms.* It was my first encounter with Seymour Papert. I found it to be one of the most engaging books I had ever read. In the book, Seymour established and developed the metaphor of the samba school as another way of looking at learning. It certainly made sense to me, especially with the exciting classroom environment of the Logo course I was taking.

Learn to dance the samba by dancing the samba! Indeed!

I can still remember the depth of the "Aha!" experience Seymour provided to me in *Mindstorms* with the simple revelation of writing a procedure named HOUSE that contained the names of other procedures named SQUARE and TRIANGLE. Suddenly, I could see all manner of teaching possibilities and educational applications dancing in my mind! This was one of the most significant moments of my life.

Shortly after that, I had the opportunity to encounter Seymour more directly. Once the National Logo Exchange (as it was called back then) was launched in September 1982, the Logo folks at the University of Virginia began to think about how to connect with even more Logo users. The idea of a Logo conference spurred a series of telephone calls to MIT, and we eventually agreed to host a planning meeting at the 1983 National Educational Computing Conference (NECC '83) in Baltimore.

Immediately prior to this meeting, Seymour and I had a telephone conversation to sketch out the idea of an international Logo conference to be held at MIT. Heady stuff, indeed, for a UVA graduate student. It was an exciting time! But for me, the dance was only beginning.

Shortly afterwards, I was lucky enough to be named to the Logo 84 conference steering committee and attended several planning meetings with Seymour and many other MIT Logo legends. I was touched by Seymour's consistent gentle demeanor and his genuine interest in others and what they were doing and thinking. He listened intently to ideas, and offered many of his own.

This was a dance of a different nature, since its context was more official and administrative in nature. At the same time, I could still see clearly the perspectives and concepts of *Mindstorms* mirrored in his ideas and guidance. I feel this underpinning was one of the keys to the success of the series of MIT conferences.

By then, Terry Cannings had demonstrated that a regional Logo conference could succeed by hosting the West Coast Logo Conference. With Logo '86 concluding the MIT international conferences, UVA and Meckler Publishing joined to host the East Coast Logo Conference (ECLC) in 1987. Naturally, we wanted Seymour as the keynote speaker.

We set up ECLC using many ideas from *Mindstorms.* Jock McLees from Terrapin taught us how to juggle. Andy David taught us how to construct a geodesic dome. Dan and Molly Watt taught us contra dancing. One of the hallmarks of contra dancing is that traditional male-female couples are not necessary. Anyone can dance with anyone else, following the Logo-like procedural instructions of the contra caller. What fun!

On the final day of the conference, I was given the opportunity to introduce Seymour for the keynote address. During my remarks, I made reference to the earlier conference activities and the contra dancing, and then said, "Last night marked a highlight of my life. I asked Seymour to dance with me . . . and he said, 'Yes!'"

In that moment, my personal metaphor of dancing with Seymour came full circle, from that initial encounter in the pages of *Mindstorms*, through the telephone calls and various conference meetings, to the opportunity to enjoy his company as a partner in a contra dance in the midst of hundreds of other Logo enthusiasts.

Looking back, I can see that dancing with Seymour has heavily influenced my life. But, more importantly, because of this dance, I have been able to pass on something of the promise, the potential, and the excitement of Logo to teachers and students all around the world. Heady stuff, indeed!

Thank you, Seymour, for saying, "Yes." I still dance with your ideas! LX

BOOK REVIEW

Reading Papert or Growing with Your Guru

by CAROLYN DOWLING

As we move through life it is customary to discard our gurus. New pre-occupations seduce us from their teachings, their inspiration no longer moves us, we've been there and done that. The guru becomes last year's news, and we move on.

So what's the matter with Papert? Why hasn't he done the decent thing by now and become boring? Why do we continue to scour the Web for the latest speeches and articles? Why do we still await forthcoming books with eager anticipation, besieging book stores for the first copies, importing them in far-flung parts of the world by all sorts of creative and devious means, passing them around as treasures from friend to friend?

There's no denying that in the circles that count (largely, but not exclusively, education), Papert is still hotter than Windows 90-anything, and seems likely to remain so.

What's his secret? I'd suggest that one reason why we don't outgrow Papert is that he has also kept on growing and moving forward. Way back when computers were first sneaking into schools, he had already traveled to the far end of the street and was peering eagerly 'round the corner, while the rest of us were still plucking up courage to follow him out of the driveway. Papert has always been ahead of the game—far enough ahead to inspire us with his vision, but not so far that the vision has seemed impossibly out of reach. His concerns are grounded in our experiences, and his solutions engage with our dreams.

We can all think of revolutionary thinkers who have had one big insight, a single powerful idea, applicable to a particular set of circumstances. Things change, and the idea is no longer so relevant, although this does not deter many gurus from continuing to voice their original message, with little modification, long after its use-by date.

This is not Papert's style, but neither is he a reed blowing in the wind of every new educational fad and fashion. His special genius has been to allow his original ideas to grow and develop, to evolve, if you like, in response to changing educational contexts, while all the time remaining true to his fundamental commitment to the creation of environments in which children are empowered to learn.

The publication of his seminal work, *Mindstorms*, with its roots in his own learning experiences and in his earlier work in the field of Artificial Intelligence, mobilized a generation of teachers who were enthused by the educational potential of computers, but were unsure as to where this lay in practice. It was the foundation stone of a whole educational "culture" possessed of a way of thinking about learning that was sufficiently distinctive, profound, and clearly articulated as to be widely dignified with the label "philosophy." Today the term "Logo philosophy" is applied far more broadly than to Logo alone, and "Logolike" as a description of learning environments is well understood outside what has sometimes been termed the "Logo community."

Subsequent work focussed on the development and refinement of a number of powerful educational ideas, including "constructionism," a realization in practical terms of key aspects of constructivist theories of learning, and the importance of epistemological pluralism—the recognition of individual learning styles and multiple ways of representing knowledge.

The Children's Machine, published in 1993, consolidated a number of aspects of Papert's thinking and writing over the previous 10 or so years. It did so in the context of an impassioned appeal for the radical reconceptual-ization and reformation of the school system as we know it—indeed, of the whole notion of "schooling."

More recently The Connected Family with its associated Web site and CD, reviewed in volume 16, no. 1 of this publication, acknowledges a potential new force in children's learning—the presence of powerful computing technology in the home. The possibility now exists to reshape the balance of power between child, home, and school in relation to learning, and to foster within the home a learning culture embodying the principles long advocated by Papert and his followers, but seemingly all but impossible to implement within existing school environments.

Papert's mind will not stand still. Things do change, but rather than diminishing the force of his thinking, change serves to create new opportunities and contexts for the expression of his vision—for he indeed is a visionary. For this we are grateful. May he keep on keeping on! LX

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

FEATURE ARTICLE

The History of Mr. Papert

by MARTIN BOYLE

ABSTRACT

This paper outlines the life and career of Seymour A. Papert. We follow the development of Papert from his early formal and informal education in the South Africa of the 1930s through to his sage era in *The Children's Machine* of the 1990s. On the way we will trace the formative influence of Papert's work with Jean Piaget in Geneva, his first serious collaborative work with Marvin Minsky, leading to the bombshell of *Perceptrons*, through to the success of his greatest educational achievement—the development of the computer programming language Logo. We will see how, in the later part of his academic career, Papert has drawn out the constructivist principles of Piaget into his own constructionism at work in the classroom. Anecdote spices the life of Papert with real humour and unexpected actions will give insightful glimpses into the workings of the great man's mind.

EARLY LIFE

Seymour A. Papert was born on March 1, 1928, in Pretoria, South Africa. The man himself seems shy or unwilling to divulge many details of his early life, though in *Mindstorms* he owns to an interest in gears from early childhood experiences, indeed:

> Before I was two years old I had developed an intense involvement with automobiles. (Papert, 1980, p. vi)

Had the world then developed as it has now such precocity may even have

enabled him to enjoy a childhood sponsored by Henry Ford!

Papert further reveals a love for Daisy, who left me with an 11th commandment:

> "Thou shall invent three theories every day before breakfast and throw them away before dinner." (Papert 1993b, p. 58)

Such is the eclecticism of Papert's subsequent endeavours that one is inclined to accept that he fell, hook with the lot, for that beloved teacher's advice.

Papert's childhood gives every appearance of being outside the norm. His father was an entomologist who spent the best part of Seymour's early childhood roaming the east coast of southern Africa in pursuit of the tsetse fly—a lifestyle that required all members of the family to turn their hands in whatever direction was required and, for a young boy, must have been more than just compensation for an eccentric upbringing.

Indeed:

> The Papert family's way of life was straight out of a Hemingway story. (Crevier, 1993, p. 84)

The story tells of following bush trails, hunting for food, and falling in love with the transmission differential of broken-down trucks—strong formative experience for the young mind.

By the nature of things, the Paperts were the only white people in the area. This led the young Seymour into trouble when he had to eventually attend school in Johannesburg. Unfamiliar with the strict and convoluted edicts of the political and social consequences of apartheid, the 10-year-old organized evening lessons for the illiterate black domestic servants of his area and found himself in serious trouble with the authorities for such illegal activities; the consequences were to have interesting future reverberations:

> This was just the first of Papert's anti-apartheid activities, activities that would later lead the United States immigration authorities to deny him a visa for many months. (Crevier, 1993, p. 84)

The logic of the situation was lost on the young boy, because:

> Adults justified their reluctance to let blacks sit at school desks by citing fear of contagious disease. But, reflected Papert, these are the same servants who take care of babies and cook the food in the whites' homes. How can the ruling class think like that? (Crevier, 1993, p. 85)

Such interest in matters logical and illogical soon earned the talented boy an invitation to attend seminars in philosophy at the University of Johannesburg!

Early Career

Before long Papert was a philosophy student in his own right at the University of Witwatersrand where he was awarded his BA in 1949. After this his interests switched to mathematics, leading to a doctoral degree from Witwatersrand in 1952.

In order to widen his horizons, Papert then moved overseas. He was awarded a Commonwealth research scholarship to St. John's College, Cambridge, UK, which would eventually enabled him to complete a second doctorate and while in England he ran into one of Americas foremost workers in the emerging field of Artificial Intelligence, Ed Feigenbaum, at the National Physical Laboratory outside London.

> Feigenbaum was a Fulbright Fellow for the year, and he had a memorable friendship with a young and somewhat eccentric South African scholar named Seymour Papert. (McCorduck, 1979, p. 275)

More significantly for the future, at a symposium in London itself, he first met Marvin Minsky. The encounter with Minsky was to prove the genesis of the second great collaborative effort of Papert's professional life; but first he turned his interests toward France.

Papert spent the year 1956-57 as a researcher at the Henri Poincare Institute at the University of Paris to complete the research for his doctorate, but then an opportunity opened and Papert was to spend a fascinating episode of his life as a researcher at the International Centre of Genetic Epistemology, at the University of Geneva, working under Jean Piaget.

> . . . the Parisian discovery that had the biggest impact on my life was Jean Piaget, who at that time was giving a course at the Sorbonne. I got to know him and was invited to work in his center in Geneva, where I spent the next four years and became passionately interested in children's thinking. (Papert, 1993b, p. 33)

Although Papert was to deviate from the gospel of the master in terms of the rigidity of the stages of child cognitive development stages, the Piagetian influence pervades the remainder of his work. Piaget, fundamentally and overtly, contributing to such diverse concepts as the establishment and efficacy of computer-rich microworlds through to the layering development of societies of mind.

Allow Papert, with metaphor as ever his telling servant, to make his own eloquent point:

> When Piaget is poured into a new decade, much will change. Whether one has conservation of Piaget will depend on what one perceives as most important in the thinking of the great master. My own view is that the essential aspects of his work have not fallen by the wayside. On the contrary they are stronger and more relevant than ever. (Papert, 1988, p. 3)

And later:

> In my Piaget, stages and even most stages of "active learning" are quite secondary. I focus instead on his constructivism and structuralism. (Papert, 1988, p. 4)

The developing influence, within the constructionist movement in education of Microworlds, StarLogo, Lego robots, and magic bricks, a decade on, strengthen that assessment of Piaget's worth.

The Artificial Intelligentsia and Perceptron Sagas

> Following the four years in Geneva, I became a professor of mathematics at MIT. Many factors made the move attractive. There was the prospect of access to computers and of working with Marvin Minsky and Warren McCulloch, as well as a wonderful sense of playfulness that I had experienced there on brief visits. When I finally arrived, all this came together in all-night sessions around a PDP-1 computer that had been given to Minsky. It was pure play. We were finding out what could be done with a computer, and anything interesting was worthwhile. Nobody yet knew enough to decree that some things were more serious than others. We were like infants discovering the world. (Papert, 1993b, p. 33)

One wonders how many of us have shared that heady experience with Papert when computers are new, computers are an unknown quantity—with computers anything goes!

In 1958 Marvin Minsky and John McCarthy had founded the Artificial Intelligence Group at MIT. Minsky had not forgotten the young Papert who had made such an impression on him at their London meeting.

> ". . . for it was at the 1960 London Symposium on Information Theory organized by Colin Cherry that an event happened which changed my career path and made me follow the course that brought me here. I came to that meeting as a mathematician interested in computational ideas and Information Theory. I came there with a paper in which I had a little theorem. And what happened was the

worst nightmare of somebody coming to a meeting with a theorem.

The speaker before me announced exactly the same theorem and proved it at least as well as I did, not quite the same, but you can't get much credit for just having a slightly different proof. Now that could have been a nightmare; in fact, it turned into a great gift. That person was Marvin Minsky. Marvin and I came to the meeting with essentially the same paper and this led to a collaboration that continued for many years and is responsible for almost everything we did in the next decade and has certainly colored everything I have done since then. (Papert, 1998)

After the previously mentioned visa problems were sorted out with the U.S. immigration authorities, Papert was in. He strode into Minsky's office, sat down, and they were away—never to look back!

> Rarely had co-operation between two researchers been so productive: Colleagues no longer said "Minsky-and-McCarthy," but "Minsky-and-Papert." The two soon initiated new research programmes in the theory of computation, robotics, human perception, and child psychology. When the Artificial Intelligence Group formally became the MIT AI Laboratory in 1968, Minsky and Papert acted as co-directors. (Crevier, 1993, p. 86)

It was an exhilarating time in AI. Sane men and women with earned PhDs from highly respected seats of learning were claiming that the office thermostat was intelligent! Perhaps with hindsight and the passage of time they deserve to retain their now found anonymity!

Alan Newell and Herb Simon back in 1956 had produced software that could churn out proofs of theorems from Russell and Whitehead's *Principia Mathematica*.

> In fact the Logical Theorist discovered a shorted and more satisfying proof to Theorem 2.85 than Whitehead and Russell had used. Simon wrote this news to Lord Russell, who responded with delight. However the *Journal Of Symbolic Logic* declined to publish an article co-authored by the Logical Theorist describing this proof. (McCorduck, 1979, p .142, footnote)

Minsky was building robots; Minsky and Papert were designing vision machines; and then there was chess . . .

All day long the argument ebbed and flowed on the matter of intelligent machines. The philosopher Hubert Dreyfus published in 1965 a report for RAND called Alchemy and AI (later expanded into the book *What Computers Can't Do.*) generally panning what Dreyfus termed the artificial intelligentsia. The ensuing battle was bloody and is, indeed, far from over!

Papert wrote the paper to refute Dreyfus, also for RAND, but the attorneys would not touch it!

> It was eventually brought out as a Project MAC report with no lawsuits ensuing. (McCorduck, 1979, p. 196)

There's still great debate about who did what and who said what in the great chess debate. The fact is that a chess playing software package that we will call MacHack throughout for simplicity had been beaten by a 10-year-old boy to the delight of Dreyfus who was alleged to have claimed that chess playing computers could never beat any human player. The program was strengthened (in fact it was a different program) and somehow Dreyfus, who was a poor chess player, was persuaded to play it.

> Papert, smiling recalls, "I organised the famous chess match. That was beautiful. He was—well, it wasn't all pathetic and sad because he was quite convincing. He was going to beat it very easily. And that also said something about him, almost naive. We didn't know. About halfway through we all thought Dreyfus was going to win." (McCorduck, 1979 p 198)

Herb Simon had this to say:

> Dreyfus thought that MacHack would play bad, mechanical, nonhuman chess. But it was a wonderful game—a cliff-hanger between two woodpushers with bursts of insights and fiendish plans . . . great moments of drama and disaster that go on in such games. (McCorduck, 1979 p 199)

Dreyfus was soundly defeated. Revenge for the artificial intelligentsia was very sweet; but, as throughout history, successful houses soon turn upon themselves.

Frank Rosenblatt had been a classmate of Minsky at high school in the early '40s. In 1962 he introduced to the press with, it must be said, some fanfare the perceptron. The perceptron was a simple neural network, a model of artificial intelligence at odds with the then fashionable symbol manipulation models. Minsky had toyed with neural networks; in fact, his PhD dissertation concerned them and had dismissed their worth at that time. Thus the claims made by Rosenblatt purporting to demonstrate the learning powers of the perceptron were viewed right from the start with skepticism in the Papert camp.

The MIT faction did take the situation seriously though. David Waltz, a graduate student recalls:

> Marvin and Seymour really were interested in Perceptrons. I and a bunch of other students took a seminar from them, where the goal was to figure out as much about Perceptrons as possible. We were

> merely to explore in a methodical sense what they were capable of and what they weren't, and try to characterize them in some way. (Crevier, 1993, p. 107)

The result of all of this activity was the publication of a book by Papert and Minsky in 1969 that they called *Perceptrons*, and which demonstrated mathematically and most ably that the simple Perceptron was totally incapable of solving a wide range of important mathematical problems.

The repercussions of their book were immediate and dramatic, for Rosenblatt and his collaborators were totally unable to rebut the book's arguments. Neural network research was dead in the water amid claims of deliberate sabotage to divert federal funding away from networks and into symbol manipulation programmes. No self-respecting researcher would dare touch neural network research for a decade until the connectionist movement of the '80s, which has proved greater potential for fruitful results. Connectionist researchers in AI blame Papert and Minsky to this day for the decade of neglect!

In the 1972 printing of *Perceptrons*, there is a handwritten dedication to the memory of Frank Rosenblatt who was lost in a boating accident, by all accounts a broken man, shortly after the Perceptron affair.

We will leave the last word on the matter to Papert; he wrote in 1988:

> Did Minsky and I try to kill connectionism, and how do we feel about its resurrection? Something more complex than a plea is needed. Yes, there was some hostility in the energy behind the research reported in *Perceptrons*, and there is some degree of annoyance at the way the new movement has developed; part of our drive came, as we quite plainly acknowledged in our book, from the fact that funding and research energy were being dissipated on what still appear to me (since the story of new, powerful network mechanisms is seriously exaggerated) to be misleading attempts to use connectionist methods in practical applications. But most of the motivation for *Perceptrons* came from more fundamental concerns many of which cut cleanly across the division between networkers and programmers. (Papert, 1992, p. 346)

Mindstorms and Logo

We turn now to that most famous episode of Seymour Papert's educational life—the development of the computer programming language Logo. We will mention only very briefly matters of common knowledge. Suffice to say that the language had its beginnings in Papert's thoughts of the late '60s and culminated in the publication of his seminal work, *Mindstorms—Children, Computers and Powerful Ideas*, in 1980.

The Logo Language was an offshoot of the list-processing language known as LISP and its genesis from lists gave rise to its name—the Greek for word, Logos. The famous turtle probably finds its antecedents in a similar living model that used to roam the Bristol UK laboratory of brain physiologist Grey Walter in search of food by way of power outlets! The rest, as most would say, is history.

A second edition of Papert's book *Mindstorms* was published in 1993 and it is to the introduction of that edition to which we turn our attention.

As is right and fitting, Papert uses the introduction to the second edition as a debugging exercise for the first edition, practising his preaching on not getting things right the first time!

Papert acknowledges that he failed to anticipate the pick-up rate of *Mindstorms* by teachers—especially elementary teachers—and is disappointed that many of his exemplary microworlds are classically physical and an impediment to such readers going beyond the early part of the work where the field is rich in Newtonian physics! Papert suggests that he would reorganize the work to present first the Images of a Learning Society to entice the elementary reader further into the plot.

Papert is also concerned with the notion, which he suggests some have read into the work, that computers will cause changes in the way children think. He replies:

> What I was saying, and still say, is something slightly more subtle: I see Logo as a means that can, in principle, be used by educators to support the development of new ways of thinking and learning. However, Logo does not in itself produce good learning any more than paint produces good art. (Papert, 1993a, p. xiv)

The emphasis in the first edition on structured programming is also regretted. Papert would much have preferred to introduce the tinkering bricolour as an alternative programming style much earlier in the piece and he stresses his subsequent work in this field with his then wife, the techno-psychologist Sherry Turkle. (Turkle, 1984).

The recent developments in Logo, in particular the MicroWorlds, are seen as a move toward the encouragement of a richer, wider epistemological range and, we may say, a more expansive interpretation of activities that constitute computer programming.

Constructionism and *The Children's Machine*

In recent years the writings of Seymour Papert return ever more strongly to his Piagetian intellectual roots. In the late '70s, his collaborative work with Marvin Minsky on the development of Society of Mind concepts is significant, though noticeably unreported on Papert's part. Such societal metaphors show, in their layering of different bands of cognition in the developing mind, an echo, of Piaget's stages of readiness. In the '80s, Papert's work with Sherry Turkle on epistemological pluralism and with Idit Harel (Harel, 1991) on children as software designers carries forward the constructivist notions of building knowledge structures into constructionist principles which are driving children's activity in this decade.

Papert would, I suspect, consider a definition of constructionism as an oxymoronic concept. We will have to do with the banal, flat, and constrained learning by doing—especially physically making something.

> Constructionism shares constructivism's connotation of learning as "building knowledge structures" irrespective of the circumstances of the learning. It then adds the idea that this happens especially felicitously in a context where the learner is consciously engaged in constructing a public entity, whether it's a sand castle on the beach or a theory of the universe. (Papert, 1991, p. 1)

Constructionism finds a true home in a computer-rich culture and herein lies the heart of Papert's objection to current educational practice. He is not as might have appeared from *Mindstorms* anti-teacher; rather, he is against the prevailing school culture that constrains children, physically and epistemologically, in the pathway of its own liking.

In his closing address to the 1990 World Conference on Computers and Education, he appealed, in the spirit of those times as a comparison with the political structure of the then USSR, for perestroika in epistemological politics.

As Papert says:

> His [Mikhail Gorbachev] slogan of perestroika (which literally means "restructuring") became synonymous with a policy of struggling to reform a system in serious crisis without calling into question the foundations on which it was built. It should be clear by now that I see most of those who talk loudly about "restructuring" in education in much the same light—though few of them have the courage to carry the reforms as far in their realm as Gorbachev did in his. (Papert, 1993b, p. 206)

Perhaps the computer is *The Children's Machine* and the vehicle for freeing thought.

Endnote

We end with this vision of Papert:

> Absent-minded like many driven intellectuals, Papert is said to have once realised, mid-way across the Atlantic, that he had left his wife behind in a New York airport. Colleagues report that he sometimes forgets to show up at lectures and, when he does, tends to get carried away into whatever topic fascinates him at the moment. A man of dramatic personal magnetism, he is likely to startle interviewers with juggling demonstrations at airport terminals or by stopping his car in the middle of a U-turn to formulate a thought. Papert's aphorisms, like Minsky's, tend to stick. One of his favourites is that we are to thinking as the Victorians were to sex. (Crevier, 1993, p. 86)

I will leave you to unwrap that saying of Papert's for yourselves!

References

Crevier, D. (1993) *Artificial Intelligence*, New York: Basic Books.

Harel, I. (1991) *Children Designers*, Ablex: Norwood, NJ.

McCorduck, P. (1979) *Machines Who Think: A personal Inquiry into the History and Prospects of Artificial Intelligence*, San Francisco: W. H. Freeman & Company.

Papert, S. (1980) *Mindstorms: Children, Computers and Powerful Ideas*, New York: Basic Books.

Papert, S. (1988) 'The Conservation of Piaget: The Computer as grist to the Constructivist Mill' in Forman, G. & Pufall, P. (Eds.) *Constructivism in the Computer Age*, Hillsdale NJ: Lawrence Erlbaum.

Papert, S. (1991) 'Situating Constructionism' in Harel, I. & Papert, S. (Eds.) *Constructionism*, Norwood, NJ: Ablex.

Papert, S. (1992) 'One AI or Many?' in Beakley, B. & Ludlow, P (Eds.) *The Philosophy of Mind*, Cambridge MA: the MIT Press.

Papert, S. (1993a) *Mindstorms–Children, Computers and Powerlful Ideas*, Second Edition, New York: Basic Books.

Papert, S. (1993b) *The Children's Machine –Rethinking School in the Age of the Computer*, New York: Basic Books.

Papert, S. (1998) The 11th Colin Cherry Memorial Lecture on Communication, London: **www.connectedfamily.com/frame4/cf0413seymour/recent_essays/cf0413_cherry_1.html**

Turkle, S. (1984) *The Second Self: Computers and the Human Spirit*, London: Granada.

Martin Boyle
Faculty of Education
Monash University
Victoria 3168 Australia
martin.boyle@education.monash.edu.au

MR. CONDOM / Continued from Page 5

ing their time" (and their condoms) blowing balloons. But this conventional wisdom entirely misses the point. Neither Debbie nor the villagers were wasting time—by playing with the condom or the fraction they were taking these things into their heads. They were letting themselves think about them.

About the Author

Dr. Seymour Papert is fondly known as the "Father of Logo." He is a respected mathematician, educator and artificial intelligence pioneer. Papert's numerous honors include awards from the Software Publisher's Association and the Smithsonian Institution. Dr. Papert occupies the LEGO Professor of Learning Chair at the MIT Media Lab's Epistemology and Learning Group. Seymour Papert's books including *Mindstorms: Children, Computers and Powerful Ideas*, *The Children's Machine*, and *The Connected Family* are required reading for anyone interested in educational computing and the future of learning.

FOR BEGINNERS

Simple Probability as a Vehicle for Understanding MicroWorlds Data Structures

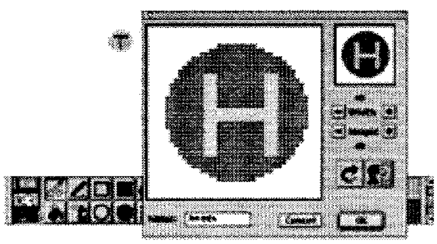
An Activity for Advanced Beginners

by GARY S. STAGER

Now that most of you can be considered advanced beginners in using MicroWorlds, this activity will explore a bit more of the language and data structures. This activity will explore probability while demonstrating how sliders, text boxes turtles and even the screen may be used to collect and report data. All of these objects are data structures in MicroWorlds

The core of this project will flip a coin numerous times and record number of times heads and tails appear.

1. Start a new project.
2. Name the turtle, *coin.*
3. Create two coin shapes in the shapes center. Name one *heads* and the other *tails.* Be sure to make them appear different in some way so that the user can clearly see which one side of the coin lands face-up.
4. Change the turtle's costume to one of the coin shapes.



5. Create a *Many Times* button with the instruction, flip.

Recording data with text boxes

This part of the project will flip a coin in FLIP, and change the value in the textboxes, headscount, tailscount and totalflips. If you name turtles, text boxes or sliders with you unique names, you may change them even if they are on different pages. This allows you to have some action going on between the scenes.

1. Make a *startup* button on the first page.
2. Create a new page from the pages menu.
3. Create text boxes named, *Headscount, Tailscount* and *Totalflips.*
4. Show the names of the text boxes so the user knows what they are reading.
5. Click the *Startup* buttonType the following procedures on the procedures page.

```
to flip
ifelse coin = "heads
[recordheads] [recordtails]
settotalflips headscount +
tailscount
end
```

The flip procedure could also be written:

```
to flip
ifelse 1 = random 2
[recordheads] [recordtails]
settotalflips headscount +
tailscount
end
```

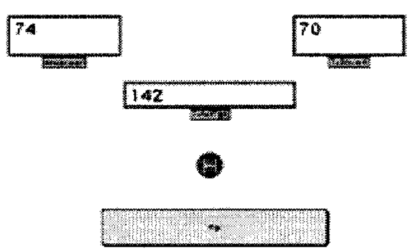
Do NOT use more than one of the flip procedures in your project!

```
to coin
if 1 = random 2 [output
"heads]
output "tails
end

to recordheads
coin, setsh "heads
setheadscount headscount + 1
end

to recordtails
coin, setsh "tails
settailscount tailscount + 1
end

to startup
setheadscount 0
settailscount 0
end
```



Click the *flip* button to start and stop the experiment. You may wish to make the *flip* button run Many Times if you want it to keep flipping the coin.

Can you make a text box that reports the standard deviation of the coin flips?

Recording data with sliders

Sliders may be used as reporters (input devices) to change the value of a variable or they may be used as indicators (output devices) displaying the current value of that reporter. Let's experiment with sliders on a second page of our coin flipping project.

1. Create a new page from the Pages menu.
2. Create two sliders, *heads* and *tails*, with a minimum of 0 and maximum of 300 at the bottom of the new page.
3. Optional: Create buttons to switch between the two pages of our project.
4. Make the following changes to your procedures.

```
to recordheads
coin, setsh "heads
setheadscount headscount + 1
setheads heads + 1
end

to recordtails
coin, setsh "tails
settailscount tailscount + 1
settails tails + 1
end
```

Change the startup procedure as follows.

```
to startup
settailscount 0
setheadscount 0
settotalflips 0
settails 0 setheads 0
end
```

1. Type Startup to initialize the variables, click on the flip button and switch between pages.

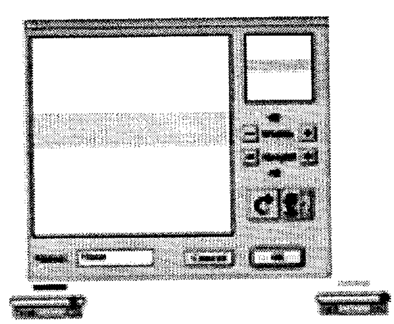
Do you see the sliders changing their values?

Extra bonus!

Adding a histogram to graph our data

It is easy to add simple graphing functionality to our probability lab with the creation of two turtles and a bit more Logo programming.

1. Hatch two turtles on the same page as the sliders.
2. Name one turtle, *headsgraph*, and the other, *tailsgraph*. (for heads graph and tails graph)
3. Place those turtles above their respective sliders.
4. Create two different turtle costumes consisting of blue and red horizontal bars. Name the shapes *hline* and *tline*.



5. Make the following changes to your procedures.

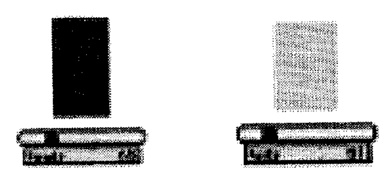
```
to recordheads
coin, setsh "heads
setheadscount headscount + 1
setheads heads + 1
headsgraph, fd 1 stamp
end

to recordtails
coin, setsh "tails
settailscount tailscount + 1
settails tails + 1
tailsgraph, fd 1 stamp
end

to startup
settailscount 0
setheadscount 0
settotalflips 0
settails 0 setheads 0
headsgraph, setpos [-170 -
145]
tailsgraph, setpos [200 -145]
page2 clean page1
end
```

1. Type Startup and click on the flip button to set the experiment in action!

You may even want to figure out a way to stop the graphing when a bar reaches the top. How about a textbox reporting the experimental standard deviation?



The magic of MicroWorlds' parallelism allows the coin to be animated, text boxes to change, sliders to report and a histogram to be created all at once. You can use lots of software to generate random numbers, but no other title allows all of these things to happen at once. I am confident that you can figure out exciting ways to integrate these programming techniques into much more complex simulations and experiments.

Gary Stager, Editor-in-Chief
logoexchange@stager.org

Dreams, Realities, and Possibilities

by GAIL MARSHALL

Introduction

Sometimes, on waking from a deep sleep, you can recall fragments of a dream: "First this happened and then suddenly that happened, but how did the end of the dream fit in with what went before?" We often have trouble making sense of our dreams because they are made up of bits and pieces of events—some real, some imagined, or changed beyond recognition in our consciousness or unconscious states. Looking back at the history of Logo in American schools is, in some ways, like a dream. We can reconstruct the impetus for introducing Logo, mark the milestones on its implementation in one or another school but then the picture becomes clouded. "What happened?" we ask. "How did the flow of events go from recognizing that Logo was a powerful tool that allowed students to use computers as mediums of expression to the present day, where, for the most part, computers are driving students?"

The Dream

Even before yesterday's primitive desktop computers appeared in American classrooms, Seymour Papert and other visionaries had the dream that computers could be used to free students from the constraints of traditional educational media and could serve as the impetus for students' engagement with important ideas, strategies and products. Logo broke the mold. No more would students be driven by the linear flow of text on a printed page; no longer would they respond to pre-programmed questions in workbooks; no longer would they go through tedious manipulations with protractors or blocks or rods to create new realities. Instead the student would be given tools that would challenge, stretch, excite and strengthen the mind.

But, for the most part, the dream was rejected by American schools. In spite of the enthusiasm of teachers, students and parents who saw Logo as a powerful stimulus for children's intellectual development, many educators resisted change and resisted possibility of using computing power to change schooling for the better.

The problem lies not in Logo and not in the stars-undefined forces of fate pushing us this way and that-but in the conflicts over what to teach and how to teach that have made education a quiet but bloody battleground for decades. For all its ingenuity, for all the possibilities that Logo provides to open up the world of mathematics and art and computing and general thinking Logo's potential on the wide scale it deserves-used as a matter of course in classrooms around the country-has not been reached because a mindset dominates American education that is essentially inimical to change, to education as a constructive act, and to the empowerment of students as agents in their own learning. As a bold, creative, purposeful endeavor Logo will have a life of its own for years to come. Fortunate students will work with Logo, realize the power that comes from working with problems they generate themselves and profit from the realization that education can be a constructive act. Wise teachers will use Logo as a means of freeing students from the routine, the hum-drum, the merely average and stimulate students to plan, to think, to act and to analyze in ways that are difficult, if not impossible, in other educational contexts. Thoughtful school administrators will understand that Logo use achieves many, if not all, of the goals deemed important by professional educational organizations such as NCTM, NCTE, ASCD, and by policy groups, both in government and the business world. But if Logo is to achieve more widespread use than it now enjoys, changes must occur. Changes not in Logo itself but in the way Americans-policy makers, parents, researchers, teachers and administrators-view education.

The Reality

Some say that American education started with Horace Mann or George Peabody sitting on one end of a bench and a student sitting on the other end. In truth, a variety of forces have shaped American education. Some of those forces were based on Romantic, Rousseauian, ideals about how students learn; others were based on instrumentalist goals-teaching vast numbers of students, some of whom had only the day before arrived from other, non-English-speaking countries to read, write, count, and move quickly into factories and foundries.

Callahan (1962) recounts the depressing story of a great shift from the quality of instruction as an issue to the quantity of services delivered by schools. During the 1920s and 1930s the "Taylor" movement, based on the ideas of the efficiency expert, Frederick Taylor, changed the face of American schools as the formula for deciding on the worth of schools was decided by dividing the number of students graduating (the "output") into the total cost of schooling (the "input"). Schools across the country were advised to drop the teaching of foreign languages when only small numbers of students in any school elected to take those courses. That such moves were penny wise and pound foolish was demonstrated years later when the federal government had to pour millions of dollars into emergency training of foreign language teachers during the Sputnik "crisis." But the underlying problem, the use of "input/output" ways of evaluating education, has persisted. As a result, American education has been driven by utilitarian goals-how many students are graduated, how many of those students get jobs in factories or foundries, how many remain in their jobs? Such utilitarian goals are inconsistent with many of the educational innovations designed to generate higher levels of student thinking, promote mathematical thinking beyond low level computation and increase humanistic values (Marshall, 1994).

The situation is complicated by the major role that testing plays in decision making. Today's standardized achievement tests are nothing more than elaborations of the types of sorting instruments designed during WWI to ensure that recruits for the Army could read and write. Easy to administer and quickly scored, the tests identified draftees who couldn't read and streamed them away from induction. After the war the growing testing industry used the models and methods of Army-driven testing for a wide range of assessments-industry used some measures for job placement and counseling, schools used them to sort out students deemed acceptable for higher education from those judged suitable only for low level jobs. By the late 1930s the acceptance of standardized tests was widespread and the "fill in the blanks/check the correct box" mode of assessment reigned supreme as a way of telling students, teachers and the community at large whether Johnny and Jane could read.

While the tests were efficient and were effective in that they provided a quantitative measure of one student against another or all students against a pre-set standard, the tests were reductive. What was judged to be important was what could be counted and what can be counted isn't always important. The problem with American testing methods was made especially evident as the curriculum reform movement of the late 1950s and early 1960s collided head on with the problem of assessing the effects of the new curricula with tests that were judged to be "scientific" by the public at large. So ingrained was the habit of using standardized tests, and only standardized tests, as acceptable outcome measures, that the public came to believe that the use of any other kind of test was a "fudge factor" connived at by educators who were unwilling to subject their curricula to "real" tests. The problems faced by the "New Math" and other curriculum reform efforts were, in no small measure due to the public's rejection of any outcome data that didn't look and feel like the good old tests of yesterday (Marshall, 1994). The curriculum developers' cautions that those tests didn't capture the complexity of performance fostered by the new curricula was hooted at by educational conservatives, who wished to maintain the status quo.

DuBois, in a comprehensive history of the testing movement that focuses on needs and conditions for testing as well as on the traditional formulaic approach that characterizes many psychometrics textbooks, tells us the Chinese, who invented testing, were able to change their tests as their civilization changed. Surely, as we move from the foundry and factory world of earlier days to the investigation of information world of the future, we can also begin to make a case for changes in what and how we test (Marshall, 1999).

While many of the new curricula probably deserved to fail others deserved to flourish and become part and parcel of the American educational experience. Remnants of some of the projects persist-indeed some of the early innovative software developed in the late 1970s and early 1980s owes much to the ideas developed through earlier curriculum development efforts. But the public's resistance to changes in teaching and testing has been a constant. Such was the climate when Logo was introduced.

The situation was made even more problematic by the methods usually used in the conduct of research on the effects of instruction. During the 19th century men of science looked for ways to explain numerically how things and people were alike or different. Some scientists filled the skulls of members of different ethnic groups with beans or stones in an attempt to prove that Caucasians were more intelligent than other groups. Agriculturists sought a mathematical shorthand to explain how the ears of corn grown in one plot of land were superior to those grown in another plot of land. As the fledgling research industry in America began to measure first corn and then students' progress through the school years, the methods of earlier scientists were adopted wholesale. Like the standardized tests used without question in schools, the pre-/post-test method of looking for effects became a standard-never mind that such methods focused on a limited set of questions and those questions not necessarily the ones to ask when looking for what students know.

The research methods popularized as a means of asking if instruction had made a difference were rooted in a behaviorist epistemology that asked how much students learned or how long it took them to learn. The possibility of an alternative epistemology, a constructivist epistemology that asked how students acquire knowledge, how they apply it over a wide range of situations, and what strategies are needed to help in the acquisition and use of knowledge, was either ignored or derided within the educational community at large (Marshall, 1993). Once again, as with testing and curriculum reform, resistance to any other method was offered as a means of maintaining the status quo-never mind that the status quo might not be useful now or in the future.

The Possibilities

Just because something has been done a hundred or a thousand or a million times doesn't mean that the way it's supposed to be. But change is difficult, especially today in the contentious atmosphere that cloaks everything from human behavior to the design of computer interfaces in an "either/or" set of conditions. To bring about change calls for strategies that go beyond the "You're wrong, we're right" chanting that quells the advance of new models and methods. In short, like it not, changes in educational policies and practices will probably only occur when and if constructivists take on the task of accumulating evidence, telling the stories presented by the evidence, and making a coherent case for alternatives in curriculum, testing and research.

Here's a scenario that might make a convincing case for change-but the scenario needs all the pieces working together in order to make the convincing case that reassures the doubting and advances the use of educational methods that differ from the "fill in the blanks" or the "cut and paste multimedia" that have captured educational technology.

In the first place, applications of Logo in the classroom as Logo was meant to be need to be identified. The conditions that gave rise to good implementation-is it specific staff development practices, is it time and modeling and support and a belief in the worth of the outcomes-need to be analyzed. The types of implementation that foster student interest and engagement need to be studied. Stories of how one goes from adoption to implementation need to be collected, analyzed and disseminated.

Similarly, we need to decide, in realistic terms, what we expect Logo use in students to produce. Is it reasonable, for example, to expect that first graders using Logo for only a few minutes a day a few weeks in the school year learn "planning skills?" Or is it reasonable to expect that students who work with Logo, in the ways Logo was designed to be used, grow more confident about asking mathematical questions and more skilled in answering those questions? If we ask such questions we also need to be able to demonstrate that acquiring such skills has positive benefits. Maybe students with a solid foundation in Logo do as well or better than their peers in algebra or geometry or calculus-or maybe they elect to take more mathematics courses or maybe they become more fluent in spatial relations. A wealth of well-designed tests, Raven's Progressive Matrices comes to mind, are regarded within education and the world at large as reliable, valid indicators of intellectual performance. Those tests and others we may need to design, subject to rigorous scrutiny and apply in order to answer the question, "What difference does Logo make?" can assuage the public's doubts about the worth of Logo's possibilities as a permanent part of a school's instructional program.

Above all, we need the students' performance to tell the story. Videos of students engaged for long periods of time with complex problems can be compelling evidence for people whose only knowledge of Logo is "It's new and it isn't computation." Presentations of the types of innovative spatial, numerical and logical thinking that occurs in the best Logo classrooms may go a long way towards convincing the dads and moms who never met an Instructional Learning System they didn't like that Logo is not a toy.

We need teachers helping other teachers in a systematic way. We need researchers who frame questions that are appropriate for the Logo environment and not researchers who are bent in applying research questions that may fit their own epistemologies but have nothing to do with Logo. Logo users need to pose the questions and not leave the generation of research to people with little understanding of children or mathematics or the possibilities for technology in the 21st century.

Seymour Papert and others who have brought Logo and Logo-like materials to market have done 99% of the work. The 1% remaining, however, is the critical percent that will ensure that more students have an opportunity to go beyond mere key punching and use technology as a means of stretching the mind, engaging the imagination and satisfying the quest for dissonance that Piaget tells us is built into every organism. It's worth the effort, don't you think?

References

Callahan, R. (1962). *Education and the Cult of Efficiency: A Study of Social Forces that Have Shaped the Administration of Public Schools.* University of Chicago Press, Chicago.

DuBois, P.H. (1967). *The History of Psychometrics.* Washington University, St. Louis, MO.

Marshall, G. (1999). Exploring assessment, (in press, Educational Information Technology).

Marshall, G. (1994). Cautionary verses: Prospects and problems in achieving the aims of the computer revolution, in J. Wright and D. Benzie, (eds.) *Exploring a New Partnership: Children, Teachers and Technology.* Elsevier Science, Amsterdam.

Marshall, G. (1993). Informatics and changes in learning: The American dilemma and opposing epistemological perspectives and unanswered questions, in D. C. Johnson and B. Samways (eds.) *Informatics and Changes in Learning.* Elsevier Science, Amsterdam.

Gail Marshall
74055.652@compuserve.com

IN THEIR OWN WORDS

Mixed Feelings about Seymour Papert and Logo

by KEN KAHN

Seymour Papert and one of his ways of describing Logo changed my life. I recall a Logo group meeting in 1977 where Seymour described Logo as an attempt to take the best ideas from computer science and make them accessible to children. Most of those ideas had come from the Lisp programming language. It felt good to hear that since I felt as if I was continuing this tradition by making a child-friendly programming language called Director. Director was inspired by the ideas of Carl Hewitt on actor programming (roughly concurrent object-oriented programming). Little did I anticipate that over the following 20 years I would make three more attempts to "child engineer" advanced programming languages.

In 1980 I left MIT and drifted away from Logo and the like. I began working on Prolog and other logic programming languages. Cynthia Solomon, who by 1982 was the director of the Atari Lab in Cambridge, was excited about the possibilities of kids and logic programming. While consulting for Atari, I did research on making the Prolog programming language accessible to kids.

Years later I was at Xerox PARC doing programming language research and had stopped doing research on computers and learning. Then around 1990 I heard a talk by Mitchel Resnick about LEGO Logo. I was impressed and told my colleagues at PARC about the talk. I also repeated Seymour's description of Logo as taking the best ideas in computer science and "child engineering" them. At the time I was working with Vijay Saraswat and others on new programming languages based upon concurrent constraint programming. Vijay asked why we couldn't make a "Logo of the 90s" based upon our research. That led to our research on Pictorial Janus.

Pictorial Janus is a completely visual programming language. The programs are pictures—they can even be drawn on paper, scanned in, automatically parsed, and executed. Program execution is shown as an animation of the original drawing. After showing it to a few children and non-programmers, I realized that despite its completely visual nature it was not well suited for non-experts. While Pictorial Janus was visual, it was based upon sophisticated formal abstractions. And these abstractions were hard for people to learn.

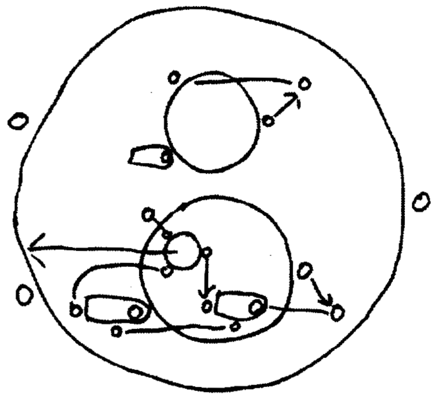


Figure 1. An Executable Hand-drawn Pictorial Janus Program

That experience led me to think of a language that was more concrete and based not upon static pictures but upon animation. And yet I wanted a language that was as powerful and expressive as the ones I had been working on at PARC. I hit upon the idea of ToonTalk (see **www.toontalk.com**), where a child does all her programming by manipulating concrete objects inside of an animated game-like world. I founded Animated Programs in 1992 to build ToonTalk. ToonTalk is a concurrent language where a child programs by training robots, giving birds messages to deliver, manipulating boxes, text pads, and number pads, using animated tools, loading trucks, and more. The child is a character in this world and can even fly her helicopter to travel between houses or to see an overview of an ongoing computation.

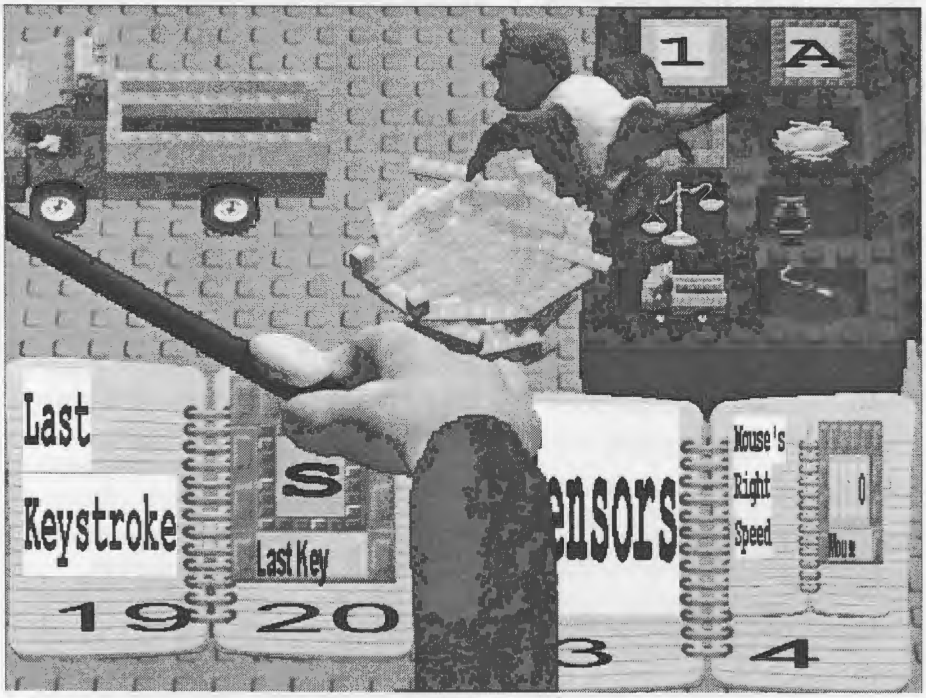


Figure 2. Programming Inside ToonTalk

So why did I title this article "Mixed Feelings about Seymour Papert and Logo"? Because I think Seymour and others did a great job designing Logo in the late '60s. But by 1975, new programming language ideas were floating around and yet Logo stayed the same. There has been great progress in programming language research in the last 25 years and yet I don't see significant changes to the Logo language that impressed and excited me 25 years ago. LCSI's MicroWorlds Logo adds some nice user interface gadgets and a very impoverished way of running programs in parallel. (Concurrent programs can't really synchronize and can only communicate via global variables.) StarLogo is more interesting but its SIMD model of computation is not flexible enough for the wide range of things that kids might want to program computers to do. It is a good thing only when dealing with problems that are naturally "data parallel."

I recall thinking in 1975, while teaching kids Logo, "Wow, here I am successfully teaching this kid a programming language that is much more advanced than what all the 'professional' programmers in the world are using." Today I look at Logo and think it isn't that different from Visual Basic and is weaker than popular languages like Java. I am disappointed that Logo and its successors are no longer based on "cutting-edge" computer science. Cutting-edge computer science research on programming languages leads to languages that are simpler, more elegant, and yet more powerful. These are good things for programmers and for kids. I still believe in Seymour and powerful ideas. I have learned a lot from Seymour and am very grateful. I just think Logo could be so much better than it is.

About the author

ToonTalk was designed and built by Ken Kahn who, after earning a doctorate in computer science from MIT, spent 20 years as a researcher in programming languages, computer animation, and programming systems for children. He has been a faculty member at MIT, the University of Stockholm, and Upsala University. For more than eight years, he was a researcher at Xerox PARC. He has made several animated films that have been shown in film festivals, theaters, and cable TV. In 1992, Ken founded Animated Programs with the mission to make computer programming child's play. He received a patent covering the underlying technology of ToonTalk (US Patent Number 5,517,663).

References

Kahn, Ken. Director guide. Technical Report 482B, MIT AI Lab, December 1979.

Kahn, Ken. A grammar kit in Prolog. In M. Yazdani, editor, New Horizons in Educational Computing. Ellis Horwood Ltd., 1984. Also in Instructional Science and Proceedings of the AISB Easter Conference on AI and Education, Exeter, England, April 1983.

Kahn, Kenneth M., and Saraswat, Vijay A., Complete visualizations of concurrent programs and their executions. In Proceedings of the IEEE Visual Language Workshop, October 1990.

Kahn, Ken, ToonTalk—An Animated Programming Environment for Children, Proceedings of the National Educational Computing Conference, Baltimore, Maryland, June 1995. Also extended version appears in the *Journal of Visual Languages and Computing*, June 1996.

Resnick, Mitch. *Turtles, Termites, and Traffic Jams: Explorations in Massively Parallel Microworlds*. Cambridge, MA: MIT Press, 1994.

Saraswat, Vijay A., Concurrent constraint programming languages. Doctoral Dissertation Award and Logic Programming Series. The MIT Press, 1993.

Ken Kahn, Animated Programs
kenkahn@toontalk.com
www.toontalk.com

StarLogo Community of Learners Workshop

by VANESSA COLELLA, ERIC KLOPFER, & MITCHEL RESNICK

New technologies can shape both what and how people learn. Before the advent of computer modeling, complex, dynamic systems were inaccessible to all but the most advanced mathematicians. The opportunity to address these ideas using computer technologies has improved their accessibility to novices and increased their presence in educational environments. StarLogo provides a way for students and teachers to build and analyze models of these systems.

But too often, people believe mistakenly that the mere presence of a new technology will be sufficient to effect change. Though StarLogo has been available online and many illustrative sample projects have been created, it has remained "unapproachable" for many people. In particular, the available materials have been insufficient to facilitate its seamless integration into classrooms. As with other technologies, the effectiveness of complex, dynamic models in education is influenced greatly by the way in which they are presented. Creating an environment that encourages and supports learners' intellectual curiosity is just as important as providing the tools for building and exploring new phenomena. Thus, we saw a critical need to develop new educational approaches for helping teachers and students use StarLogo to learn about complex, dynamic systems.

Though StarLogo has been available online and many illustrative sample projects have been created, it has remained "unapproachable" for many people.

The Community of Learners Workshop

This past summer, high school students and teachers transformed a conference room at the Santa Fe Institute into a buzzing interdisciplinary modeling workshop. The two-week StarLogo Community of Learners Workshop brought together 25 participants and five workshop leaders from around the country for a hands-on experience in model building. Participants in the workshop explored a variety of social and scientific models, including predator-prey dynamics, population growth, and traffic patterns, StarLogo2 uses, and a variety of participatory activities.

The StarLogo Community of Learners Workshop was created to help people develop new ways of understanding and describing phenomena in the world around them. The Santa Fe Institute, the Massachusetts Institute of Technology Media Lab, and the University of Maine combined forces to help high school students and teachers from a variety of disciplines learn about modeling and complexity. The workshop aimed to engage participants in building StarLogo models of complex systems and, in the process, to embrace new and powerful ideas.

Challenges

One of our primary motivations when we organized the workshop was to achieve a balance between structure and exploration. We designed the workshop to foster a playful, cooperative, creative spirit while at the same time providing adequate structure for learning how to build models. To accomplish this balance, we organized the workshop around a set of open-ended challenges and activities that guided participants' explorations and provided a means for critical review while maintaining an investigative atmosphere. Each challenge was a problem statement that was meant to get participants' creative juices flowing. Here is a short excerpt from one challenge:

> As you observe behavior in the real world, you can make guesses about the reasons for some of these behaviors. Think about the last time you were "people watching." If you saw a woman in a business suit running to catch a bus, you might have assumed that she was late for work. If

you saw a young man pacing in front of a movie theater, you might have assumed that his date was late to meet him. If you saw a small child throwing a temper tantrum, you might have assumed that he was tired or hungry. In each of these cases, you were drawing conclusions about people's internal states based upon the behavior that you observed. Often, internal states-like being hungry, anxious, happy, or energetic influence the way people act. Many other creatures and objects, even subatomic particles, are also affected by their internal states. Modeling the internal states of individuals provides you with an opportunity to understand a great variety of real-world systems.

Challenge

You have seen that turtles have several properties (e.g., heading, color, etc.) that allow them to remember information. Turtles can remember additional information about themselves if you add new turtle variables. The different types of information that turtles remember are referred to as "states."

Can you create a StarLogo project with a turtle "energy" state? How will you enable the turtles to gain and lose energy? How might your turtles behave or look differently as their energy level increases or decreases? How does their behavior help you visualize the turtles' changing states? What assumptions do you make about a turtle's state based on its behavior? Are your assumptions accurate?

The challenges facilitated model design and construction, built familiarity with the StarLogo environment, and introduced the principles of complex systems.

As we developed the challenges, we made sure they fulfilled three requirements:

1. covered a StarLogo modeling concept that was appropriate for participants' current level of experience
2. enabled participants to build a coherent and interesting model, even before they knew much of the StarLogo language
3. supported a wide variety of solutions. Often challenges contained pointers to sample projects, which illustrated possible implementations of the key concepts in the challenge. This format minimized direct instruction, instead fostering an environment in which specific programming information was available when needed.

> We adopted two key components of the design studio, the "desk-crits" and the "pinups." These mechanisms provide learners with generative feedback during the design and building processes

Many of the challenges were accompanied by "off-screen" activities that provided another way for participants to connect abstract notions of complexity and decentralized behavior to their own personal experience. One morning, participants "flew" around the parking lot trying to form cohesive "bird flocks" without a leader. Group activities like this one enabled participants to develop a personal perspective on concepts like exponential growth, local versus global information, and group decision-making. We found that integrating the group activities into the challenges was a great way to help people develop deeper understandings of complex systems. These experiences helped participants as they designed and created their own final projects later in the workshop.

Design Studio

Because we designed the challenges to catalyze a variety of possible solutions, we wanted to incorporate a structure into the workshop that would to help participants evaluate these solutions. We borrowed a few ideas from the design studio teaching model, which is standard fare in the visual and creative arts. In integrating these ideas, we hoped to foster the kinds of creative thinking that we feel should be a part of the scientific enterprise, while at the same time providing a time-tested structure for community-based learning and peer review.

We adopted two key components of the design studio, the "desk-crits" and the "pinups." These mechanisms provide learners with generative feedback during the design and building processes. While participants worked on the challenges, workshop leaders periodically visited each group. During these "desk-crits," the workshop leaders encouraged participants to describe what they wanted their models to show, reflect on how well their models achieved this goal, and think about the next steps they wanted to take. About two-thirds of the way through each challenge, "pinup" sessions provided a public forum for participants to articulate the rationale behind their models. During these sessions, special attention was paid to how well the models communicated ideas to other people. After each brief presentation, participants and leaders gave feedback to the model builders, enabling people to improve their models. Participants developed both the abilities to design, implement, and explore simulations (through the challenges) and the facilities to describe, analyze, and critique simulations (through the "desk-crits" and "pinups").

Moving Ideas Beyond the Workshop

Written evaluations from participating students and teachers showed consistent approval for the content and the format of the workshop. More important, the enthusiasm from the summer workshop has poured into participants' daily pursuits. The teachers and students continue to explore what they began at the workshop. One teacher introduced StarLogo to the Santa Fe Community College, where students are addressing local community problems, such as the declining water table. Meanwhile, more than a dozen students are building StarLogo projects for this year's New Mexico High School Supercomputing Challenge. Another teacher wrote "[my] students are very playful with StarLogo and have developed models of evaporation and cloud formation, processionary caterpillars, city rioting, and traffic flow."

We plan on running similar workshops at the Santa Fe Institute and in the Boston area during the summer of 1999. In addition, we are currently writing a workbook that includes challenges and activities inspired by the StarLogo Community of Learners Workshop. The workbook will include sample projects for each of 10 challenges, hints and strategies for completing the challenges, and ideas for presenting the challenges in a variety of learning environments. It will describe how to run "off-screen" activities that can help learners understand more about complex dynamic systems and will include a short section on why and how you might use individual-based modeling environments like StarLogo in your own pursuits. LX

> One teacher introduced StarLogo to the Santa Fe Community College, where students are addressing local community problems, such as the declining water table. Meanwhile, more than a dozen students are building StarLogo projects for this year's New Mexico High School Supercomputing Challenge.

About the authors

Vanessa Colella is a doctoral student at MIT's Media Laboratory, where she investigates how innovative computational tools can transform the processes of science education. Her research focuses on ways to help people teach and learn about complex dynamic systems. Before her work at the Media Lab, she taught junior high and high school science. She currently works on Participatory Simulations—computer-supported activities that enable immersion in and investigation about "life-sized" dynamic systems. In addition, she is coauthoring a book on StarLogo for students and teachers. She holds an S.M. in media arts and sciences from MIT, an M.A. in communications, computing, and technology from Teachers College, Columbia University, and an S.B. in biology from MIT.

Dr. Eric Klopfer is a National Science Foundation postdoctoral fellow at the University of Massachusetts at Amherst. He received his Ph.D. from the University of Wisconsin, and his B.S. from Cornell University. Dr. Klopfer's research focuses on the use of computer models in science and math education. He is currently investigating the effects of teachers developing their own computer models. In addition to the StarLogo book that he is co-authoring, he is also developing software (code-named "Weevolution") for modeling learning and evolution in animals.

Dr. Mitchel Resnick, associate professor at the MIT Media Laboratory, studies how new techological tools and media can help bring about deep changes in how people think and learn. Resnick led the development of the prototype for the award-winning LEGO Mindstorms product (a robotics construction kit based on programmable LEGO bricks), and he is the developer of StarLogo (an environment for modeling complex dynamic systems). Resnick was also co-founder of the Computer Clubhouse, a network of after-school learning centers for youth from under-served communities. Resnick earned a B.A. in physics at Princeton University (1978) and M.S. and Ph.D. degrees in computer science at MIT (1988, 1992). He is author of the book *Turtles, Termites, and Traffic Jams*, published by MIT Press in 1994.

Endnotes

[1] Vanessa Colella (MIT Media Lab), Larry Latour (University of Maine), Eric Klopfer (University of Massachusetts, Amherst), Nigel Snoad (Santa Fe Institute), and Gil Munden (University of Maine) were Workshop leaders. Special thanks also to Ginger Richardson (Santa Fe Institute), without whom the Workshop would never have happened.

[2] Developed by Mitchel Resnick, MIT Media Lab. StarLogo is available for free on the Web at **www.media.mit.edu/starlogo.**

[3] Thanks to David Shaffer for his assistance. See **http://dws.www.media.mit.edu/people/dws/papers/designstudio/index.html** for more details.

Vanessa Colella
vanessa@media.mit.edu

Eric Klopfer
eric@taumoda.com

Mitchel Resnick
mres@media.mit.edu

Papert's Conjecture about the Variability of Piagetian Stages in Computer-Rich Cultures

by ANNE McDOUGALL

Introduction

One of the most interesting and important aspects of Papert's work in Mindstorms (1980) is his conjecture that children in a computer-rich learning environment using Logo might be able to engage in learning activities involving abstract or formal thinking at ages considerably younger than would be expected from Piagetian theory.

> This book is about . . . using computers to challenge current beliefs about who can understand what and at what age. It is about using computers to question standard assumptions in developmental psychology and the psychology of aptitudes and attitudes. . . . I take from Piaget a model of children as builders of their own intellectual structures. . . . All builders need materials to build with. Where I am at variance with Piaget is in the role I attribute to the surrounding cultures as a source of these materials. . . . In many cases where Piaget would explain the slower development of a particular concept by its greater complexity or formality, I see the critical factor as the relative poverty of the culture in those materials that would make the concept simple and concrete.
> (Papert, 1980, 4-7)

From the earliest work with children and Logo, anecdotal evidence suggested that Papert's view was correct, and I became interested to see whether it could be substantiated by more formal research.

I began my investigation of these ideas with an intensive case study, the data collection for which spanned the years 1984-86. In 1988 I spent a term teaching Logo in a Grade 5 classroom to investigate the possible extension of my findings from the case study into a more generalizable classroom situation. More recently, through a project "Children as Programmers" funded by Charles Sturt University, a colleague, John Oakley, and I have been working with teachers in well resourced classrooms to investigate the conjecture further. This paper describes my earlier research on this question, and outlines the current work.

A Case Study 1984-86: Children, Recursion, and Logo Programming

For my doctoral research project (McDougall, 1988; 1990a) I undertook a case study to investigate Papert's conjecture. I studied the development of understanding and use of the concept of recursion, a topic usually considered difficult for undergraduate computer science students, in children of elementary school age. Recursion is one of the very "abstract" powerful ideas made accessible through Logo, but apart from Logo contexts, it is not normally considered appropriate for formal presentation to children at elementary school level.

The study was focused by several themes from Papert's work: the importance of the culture in which children learn in providing suitable resources for building intellectual structures, the power of computing to "concretize" abstract ideas, and the idea that learning with a computer might change the way other learning takes place.

Support of Papert's conjecture in the case study environment would not of course imply its necessary transfer to classroom settings. However, if his ideas were not supported in the study, the chances of their being substantiated in real classrooms would be very small.

The subjects for the study were my two children, aged 9 and 6, when data collection commenced in 1984. They were provided with a home learning environment rich in materials and opportunities for learning about recursion, including an Apple II computer with Logo, a robot turtle and a printer, a teacher well versed in Logo and related issues of learning and teaching, books and other materials, and, as far as possible, unrestricted time for working with these things. The learning environment was informal, with a sense of playfulness; there was no set curriculum to be covered and no assessment.

In one sense, it was not an ideal Logo culture such as Papert described in Mindstorms. The ultimate agenda for our activities was inevitably mine—the carrying out of a research project—rather than being completely the children's own. Within that constraint, however, as far as possible the children had control over our activities.

Very detailed, though not constant, observations of the children's programming and other activities in this environment were made over three years. In particular, all incidents relating to the development of their understanding of recursion were recorded. The criterion I set for substantiation of the conjecture in this case was confident and independent writing and debugging of Logo procedures using embedded recursion (Leron, 1985).

I used a peer-teaching technique for data collection. While I planned to obtain some data about the two children's learning from sessions in which they worked alone with Logo, greater insights into their learning would be possible if, having learned Logo themselves, each child were to teach it to a friend. Reports of the discussions occurring with pairs of students working together (see, for example, Hoyles et al., 1984; 1985) gave me the idea that peer teaching could be a very valuable and revealing situation for the Logo researcher. First, my children would learn what they might about Logo and recursion; then each would teach Logo to a chosen school friend in whatever way she wanted.

Seventy-nine Logo sessions were spread unevenly over the data collection period. For most of the first year, the two case-study children worked individually with me. Then for more than a year, each child worked with her friend. Finally, for two months toward the end of the third year, the elder child and her friend worked on projects particularly involving recursion.

Detailed descriptions of the children's development of understanding of recursion and skills of recursive programming have been reported elsewhere (McDougall, 1985; 1988; 1992). I'll present the briefest summary here.

The case-study children's early work with recursion did not involve programming at all. Both children recognised and made many situations involving self-reference, repetition with small variation, and nesting of levels with pictures, story fragments, toys, and everyday materials.

Since the younger child's programming work did not involve procedure writing, she did not use recursion in Logo. The elder child's first recursive programming project was the development of a tail-recursive procedure to draw a nested squares design, which she completed with help at age 9. A few months later, I wrote for her an embedded recursive Logo-like procedure for the story *The Cat in the Hat Comes Back* (Suess, 1958), and showed her a graphics procedure emphasizing the unwinding part of embedded recursion.

Her understanding of and competence in programming with recursion developed unevenly but clearly, until the third year of data collection for the study when the elder child and her friend, both then aged 11, worked on a series of recursive programming projects, showing competence, confidence and independence in using recursion in Logo. Various areas of difficulty they encountered during these projects, and their overcoming of these are described in detail elsewhere (McDougall, 1988; 1990b). The difficulties included problems with interpretation of the function of the STOP command, appreciating the scoping of variables, passing of values between procedures, and multiple inputs in recursive calls.

I shall present here two of these children's most advanced projects to illustrate the level of their programming at the end of the study.

A Graphics Project: Display Procedure for Nested Reflected Polygons

This project stretched over two sessions on consecutive days; the total time was three hours and 15 minutes. The children had previously written an ongoing "display" of spirals of different shapes, each spiral being drawn by the turtle, then erased before the next was drawn. They planned to make a similar display, built on a procedure they had already written to draw nested reflected squares.

Following are the final versions of their procedures for the display of nested reflected polygons. Some of the designs generated by running the procedure O with inputs 45 and 5 are shown in Figure 1.

```
TO O :N :T
FULLSCREEN
IF :T = 360 [STOP]
FP :N :T
REPEAT 2000 [  ]
CS
O :N (:T + 5)
SPLITSCREEN
END

TO FP :N :T
PU HOME PD
IF :N = 0 [STOP]
PU HOME PD
F :N :T
PU HOME PD
FP (:N - 15)  :T
PU HOME PD
F :N (-:T)
PU HOME PD
END

TO ABS :NUM
OP IF :NUM < 0 [ -:NUM] [
:NUM]
END

TO F :N :T
REPEAT ABS 360/:T [FD :N RT
:T]
END
```

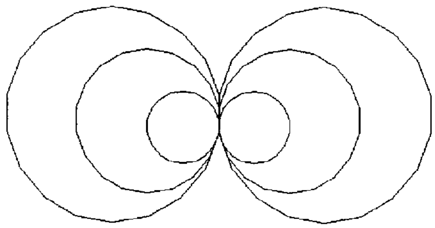


Figure 1

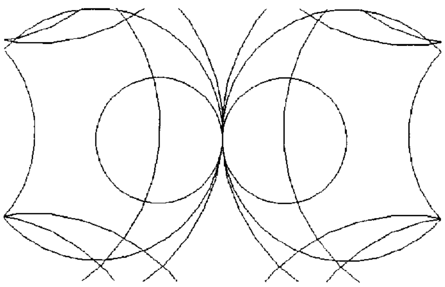


Figure 1A

Editor's translations from the ancient Logo (at left)

```
TO O :N :T
IF :T = 360 [STOP]
FP :N :T
Wait 20
CS
O :N (:T + 5)
END

TO FP :N :T
PU HOME PD
IF :N = 0 [STOP]
PU HOME PD
F :N :T
PU HOME PD
FP (:N - 15) :T
PU HOME PD
F :N (-:T)
PU HOME PD
END

TO ABS :NUM
OP IFELSE :NUM < 0 [-1 *
:NUM] [:NUM]
END

TO F :N :T
REPEAT ABS 360 / :T [FD :N
RT :T]
END
```

A List Processing Project: Too Many Bears

The elder case study child had previously read the story "Too Many Bears" (Ahlberg & Amstutz, 1983), recognised its structure as recursive, and suggested that her friend read it as well. They planned to "write the story in Logo". This project was done in one one-hour session. I helped with the IF test in the TELL.STORY procedure, suggesting that it test for [GOLDILOCKS] instead of the empty list, and with the syntax of the PRINT statements. The children's procedures for the project were the following.

```
TO CH.1
BEARS.IN.BED
QUESTION
TELL.STORY [ALLAN.AHLBERG
DINAH GOLDILOCKS]
SLEEP
END

TO BEARS.IN.BED
PRINT [BIG MEDIUM SMALL]
PRINT [DADDY MUMMY BABY]
PRINT [DINAH'S BED]
END

TO QUESTION
PRINT [???]
END

TO SLEEP
PRINT [GOODBYE]
PRINT [GOODNIGHT]
PRINT [SNORE]
END

TO TELL.STORY :L
IF :L = [GOLDILOCKS] [PRINT
[GOLDILOCKS TELLS STORY]
   PRINT [GOLDILOCKS FINISHES
   STORY] STOP]
(PRINT FIRST :L [TELLS ABOUT]
BUTFIRST :L)
TELL.STORY BUTFIRST :L
(PRINT FIRST :L [FINISHES
STORY])
END
```

Running the procedure CH.1 produced:

```
BIG MEDIUM SMALL
DADDY MUMMY BABY
DINAH'S BED
???
ALLAN.AHLBERG TELLS ABOUT
DINAH GOLDILOCKS
DINAH TELLS ABOUT GOLDILOCKS
GOLDILOCKS TELLS STORY
GOLDILOCKS FINISHES STORY
DINAH FINISHES STORY
ALLAN.AHLBERG FINISHES STORY
GOODBYE
GODDNIGHT
SNORE
```

Conclusion for the Case Study

Both the case study children, at ages 9 and 6, respectively, were able to recognize and generate examples of the features of recursion in pictures, stories and everyday situations, although they found it difficult to define or talk in general terms about recursion. The elder child, working at first (at age 9) with the researcher, and later with a friend (until age 11) developed Logo programming skills including confident use of controlled embedded recursion.

These children, in an environment rich in opportunities for working with recursion, were able to understand and use this abstract idea when one of them was as young as 6 years of age. At age 9, the elder child was able to read with understanding Logo procedures using embedded recursion and to prepare with help a procedure containing tail recursion. At age 11, she and a school friend together operated as competent and independent programmers with full recursion in Logo. Thus Papert's conjecture was supported, in the case of recursion, by this study.

My next step was to use the experience I had gained from the case study and try to test the conjecture in a classroom situation.

Work in a Grade 5 Classroom 1988

In the first half of 1988, I was able to work with Logo in a Grade 5 classroom at Balwyn North Primary School in Melbourne. There was one Apple IIe computer in the classroom, a fairly typical level of resourcing in elementary schools here at the time. I had the computer connected to a large television monitor so all the children could see what was on the screen, and I brought in a Logo disk, books, manuals, other materials, and a robot turtle for the class to use.

We worked together as one large group on directing the robot turtle to move and draw pictures. I showed the children the format for writing procedures, and we had the turtle draw geometrical figures and more complex pictures. Our projects were, I think, typical of the Logo activities undertaken in many elementary classrooms. However, with the limitations imposed by the use of a single computer for the whole class, it was quite impossible to create anything like a level of unrestricted time, and of learner ownership of our activities, sufficient to investigate Papert's conjecture.

Mulling over the unreasonableness of the arguments that "Logo didn't deliver what it promised" in environments similar to the one in which I was working, I decided to defer further investigation of the conjecture until computers were sufficiently widely available in schools for each child to have his or her own. The opportunity came with the adoption by a number of Australian schools of the use of portable notebook computers (see for example McDougall, 1995). Often these schools required that individual students own the computers, so issues of access and available computer time no longer limited what children might achieve with Logo.

The Children as Programmers Project 1996-9

With a colleague, John Oakley, from Charles Sturt University, I embarked in 1996 on a project we called Children as Programmers. With funding from Charles Sturt University, we planned to carry out further investigation of Papert's conjecture, collecting data in technology-rich elementary school classrooms where children were undertaking programming projects as part of their regular learning activities.

We chose classrooms with teachers who were highly competent and experienced in this type of work (Oakley & McDougall, 1997a) sited in schools that provided supportive environments for this type of work (Oakley & McDougall, 1997b). We interviewed the teachers about their work and that of their students.

This work is ongoing. We have a collection of disks of individual students' projects, and we are examining these systematically to investigate the programming concepts used in the projects and evidence of any other abstract concept development supported by the programming activity.

Conclusion

Papert's description in *Mindstorms* of the power of his early experiences with gears in enabling his later understanding of a variety of more abstract concepts is still, for me, one of the most exciting learning stories I have encountered. His conjecture about the potential of computing to support and enhance the learning of abstract concepts remains an extremely important one, and its investigation warrants a considerable and sustained research effort.

References

Ahlberg, A. and Amstutz, A. (1983) Too Many Bears. In *Ten in a Bed*. London: Granada. 9-17.

Hoyles, C., Sutherland, R. and Evans, J. (1984) A Preliminary Investigation of the Pupil-Centred Approach to the Learning of Logo in the Secondary School Mathematics Classroom. University of London Institute of Education.

Hoyles, C., Sutherland, R. and Evans, J. (1985) And Then She Had the Idea of Dividing It by 360. *Times Educational Supplement*. March 1, 52-53.

Leron, Uri (1985) Logo Today: Vision and Reality. *The Computing Teacher*. February. 26-32.

McDougall, Anne (1985). Teaching and Learning about Recursion. In Palmgren, M. (ed.) *Logo85* Pre-proceedings. Massachusetts Institute of Technology. 155-6.

McDougall, Anne (1988) Children, Recursion and Logo Programming. Unpublished Ph.D. thesis. Monash University.

McDougall, Anne (1990a) Children, Recursion and Logo Programming: An Investigation of Papert's Conjecture about the Variability of Piagetian Stages in Computer-Rich Cultures. In McDougall, A. and Dowling, C. (eds.) *Computers in Education*. Amsterdam: North Holland. 415-418.

McDougall, Anne (1990b) Student Difficulties in Programming with Recursion in Logo. Proceedings of the Computer Education Group of Victoria Annual Conference. Melbourne: Computer Education Group of Victoria. 108-115.

McDougall, Anne (1992) Structure and Process Microviews: Partial Understandings of Recursion in Logo Programming. In Nevile, L. (ed.) *Logo and Mathematics Education*. Melbourne: A.C.E.R. 109-118.

McDougall, Anne (1995) Integration of Portable Computers into Australian Classrooms: Issues and Outcomes. In Watson, D. and Tinsley, D. (eds.) *Integrating Information Technology into Education*. London: Chapman & Hall. 281-284.

Oakley, John and McDougall, Anne (1997a) Issues in the Preparation of Teachers of Programming for Children. In Passey, D. and Samways, B. (eds.) *Information Technology Supporting Change through Teacher Education*. London: Chapman & Hall. 108-114.

Oakley, John and McDougall, Anne (1997b) Educational Environments for Research on Cognitive Development through Programming. In Johnson, D. L. and Maddux, C. D. (eds.) *Logo: A Retrospective*. New York: The Haworth Press. 83-93.

Papert, Seymour (1980) *Mindstorms: Children, Computers and Powerful Ideas*. Brighton, U.K.: Harvester.

Seuss, Dr. (1958) *The Cat in the Hat Comes Back*. New York: Random House.

Anne McDougall
Anne.McDougall@education.monash.edu.au

Papert's Perspectives, Research Responses

by DOUGLAS H. CLEMENTS AND JULIE SARAMA

Seymour Papert has made myriad contributions to education, but perhaps none so great as providing new visions (Papert, 1980; Papert, 1987; Papert, 1993; Papert, 1996). In this brief space, we can mention only a few of these perspectives that have been investigated by researchers. What does empirical evidence say about the vision Papert has shown us?

'Spontaneously' learning mathematical ideas

One of Papert's early notions was that more ideas could be learned "preconsciously" and "spontaneously," as simple ideas about counting are learned (Papert, 1980). That is, certain processes, such as combinatorics, are not normally learned without formal teaching. However, such domains could be learned in a "Piagetian," natural way, if experiences were provided by the culture. Papert argued that the computer could provide such experiences. In support of such a notion, Wright says 4-year-olds have been seen systematically combining the primary paint colors in a computer simulation program to build the three secondary colors so that they could create a picture with all sizes and colors (Wright, 1994). We do not know how frequent such learning might be.

Learning geometry through 'body syntony'

Papert introduced turtle geometry to the earliest versions of Logo. His idea was that the turtle was "body syntonic"—firmly related to children's sense and knowledge of their own bodies and the way they move. Research supports Papert's contention that ideas of turtle geometry are based on intuitive knowledge (Kynigos, 1992). In one intensive study of the learning of one child, we found that Logo helped connect the two ideas of turn-as-body-motion and turn-as-number. Only after making this connection did the child begin to understand the measurement of angles and turns (Clements, Battista, Sarama, & Swaminathan, 1996).

Papert also argued that new knowledge acquired in Logo has to compete with existing knowledge. This notion is supported by findings that Logo has more effect on children's ideas about angles than about length (Noss, 1987). So, teachers need to be active helping children integrate what they are learning in Logo with their existing knowledge. It may be fine for children to be confused (as between the angle of turn and the measure of the resulting angle)—Papert's "transitional theories"—but then an important role for the teacher involves arranging experiences that prevent students from simply evading the problem (Simmons & Cope, 1990).

Making a transition between informal and formal ideas: The case of algebra

Papert believed that Logo could be a transition between informal and formal mathematical experiences. "When mathematizing familiar processes is a fluent, natural and enjoyable activity, then is the time to talk about mathematizing mathematical structures, as in a good pure course on modern algebra." Richard Noss (1986) tested this idea. Children who had used variables in their Logo work were more able to use algebraic ideas to represent various situations. Noss states that it is not necessary that they have learned something about algebra per se. The algebraic thinking that they learned in Logo was used to build algebraic meaning in non-computer contexts. So, Logo learning may aid in forming primitive conceptions of algebraic notions at an early age, which may become part of a system of algebraic thinking later.

Building a conceptual framework for learning mathematics

Papert (1980) claimed that Logo can make the abstract concrete, accelerating cognitive development. That is, Logo can help build a conceptual framework on which later mathematics learning can build. Using this approach, some researchers have reported gains (Miller, Kelly, & Kelly, 1988; Rieber, 1987), but others found no significant differences (Clements & Gullo, 1984; Howell, Scott, & Diamond, 1987). With teacher planning and mediation, however, computer programming can facilitate mathematics achievement and higher-order thinking (Billings, 1986; Reed, Palumbo, & Stolar, 1988; Roblyer, Castine, & King, 1988; Wiburg, 1989).

So, research reports that the use of Logo can afford opportunities for children to explore mathematical ideas in a meaningful context and in doing so generate a more active exploration and understanding of concepts and processes (Yelland, 1995). Such benefits occur more frequently and more powerfully when teachers are actively involved mediating students' work with Logo. In general, there is research support for Papert's "power principle"-students use the mathematics first for a personally meaningful goal, then deepen that initial understanding (Papert, 1996).

Learning powerfully through projects

This power principle is consistent with another perspective: Papert strongly advocated learning through in-depth engagement in projects. Here there is strong research support (with the caveat that they are all from Papert's group). Whether they designed projects to teach others about fractions (Harel & Papert, 1990) or mathematical games (Kafai, 1993), students learn more and learn more deeply, when engaged in projects.

Learning responsibly and learning together

Papert also predicted that Logo could help children take responsibility for their learning and work together more powerfully. Logo experiences positively affect students' (both boys and girls) internal locus of control (Bernhard & Siegel, 1994). Overall, research suggests that educators build classroom cultures that encourage students to take responsibility for their own learning; to engage in tasks that are challenging, but not too difficult or too easy; and to work cooperatively, asking each other questions (King, 1989), engaging in cognitive conflicts, and always working to resolve them through discussion of ideas and negotiation (Clements & Nastasi, 1988; Hoyles, Healy, & Sutherland, 1991). This should not be taken to mean that children should always work together, however; a balance of cooperative and individual work may be ideal. A combination of structured interdependence and individual autonomy, with a high-status student coordinator, may be best (Hoyles, Healy, & Pozzi, 1994).

Learning and teaching

Many researchers interpreted Papert's early writings as indicating no need for teaching. One researcher for example, used "a 'standard' play-oriented version of the LOGO curriculum as described in most published accounts (e.g., Papert, 1980)" (Mitterer & Rose-Drasnor, 1986, p. 178). This researcher found very little Logo-specific change in problem solving behaviors, however, leading to the conclusion that learning Logo will not enhance general problem solving skills.

Interpretation is individual. I, for example, can not read about the Samba schools without imagining learning and teaching beyond simple versions of play. Nevertheless, as we discussed previously, "learning without being taught" is also a Mindstorms theme. The definition of teaching is critical: teaching-by-telling in a narrow sense or guiding learning in the broadest sense?

Researchers consistently report that the most positive results have involved teacher mediation based on a well-developed theoretical foundation (Clements, 1990; De Corte & Verschaffel, 1989; Delclos & Burns, 1993; Lehrer, Guckenberg, & Lee, 1988; Lehrer, Harckham, Archer, & Pruzek, 1986; Littlefield et al., 1988). Effective teachers appear to plan and oversee computer programming experiences to ensure that students reflect on and understand the mathematical concepts (McCoy, 1996, p. 443). Research indicates that they:

- select or create tasks designed to achieve educational goals
- focus students' attention on particular aspects of their experience
- educe informal language and provide formal mathematical language for the mathematical concepts
- emphasize planning for algorithm development
- suggest paths to pursue
- provide metacognitive prompts and ask higher-order questions
- facilitate disequilibrium using computer feedback as a catalyst
- provide tailored feedback regarding students' problem-solving efforts
- discuss errors and common misunderstandings
- continually connect the ideas developed to those embedded in other contexts
- provide modeling and coaching
- promote both student-teacher and student-student interaction.

It is also noteworthy that positive effects also take a considerable time. Liu (1997) found 150 hours of experience was necessary. In the case of algebra, for example, the extent to which the researcher prompted for a Logo connection were instrumental in students' generating meaningful notations for the unknowns (Noss, 1986).

Changing schools

Papert stated that the computer by itself was not an agent of change (a perspective he called "technocentrism", Papert, 1987). This has been clearly supported by research. It is less the inherent features of computers and more people's experience with the machine that determines learning and social change (Mehan, 1989).

Final Words

This is not, and could not be, a comprehensive review of Papert's perspectives or research responses. It is but a small selection illustrating the power of Papert's vision and the empirical support-and enrichment-that research has offered. We need additional research across the spectrum-from the traditional hypothesis tests that will evaluate the many (usually post hoc) conjectures and initial results in the literature to exploratory studies that stretch the conceptions of what computer programming and education might be.

References

Bernhard, J. K., & Siegel, L. S. (1994). Increasing internal locus of control for a disadvantaged group: A computer intervention. *Computers in the Schools, 11*(1), 59-77.

Billings, L. J., Jr. (1986). Development of mathematical task persistence and problem-solving ability in fifth and sixth grade students through the use of Logo and heuristic methodologies. Dissertation Abstracts International, 47, 2433A.

Clements, D. H. (1990). Metacomponential development in a Logo programming environment. *Journal of Educational Psychology, 82*, 141-149.

Clements, D. H., Battista, M. T., Sarama, J., & Swaminathan, S. (1996). Development of turn and turn measurement concepts in a computer-based instructional unit. *Educational Studies in Mathematics, 30*, 313-337.

Clements, D. H., & Gullo, D. F. (1984). Effects of computer programming on young children's cognition. *Journal of Educational Psychology, 76*, 1051-1058.

Clements, D. H., & Nastasi, B. K. (1988). Social and cognitive interactions in educational computer environments. *American Educational Research Journal, 25*, 87-106.

De Corte, E., & Verschaffel, L. (1989). Logo: A vehicle for thinking. In B. Greer & G. Mulhern (Ed.), *New directions in mathematics education* (pp. 63-81). London/New York: Routledge.

Delclos, V. R., & Burns, S. (1993). Mediational elements in computer programming instruction: An exploratory study. *Journal of Computing in Childhood Education, 4*, 137-152.

Harel, I., & Papert, S. (1990). Software design as a learning environment. *Interactive Learning Environments, 1*(1), 1-32.

Howell, R. D., Scott, P. B., & Diamond, J. (1987). The effects of "instant" Logo computing language on the cognitive development of very young children. *Journal of Educational Computing Research, 3*, 249-260.

Hoyles, C., Healy, L., & Pozzi, S. (1994). Groupwork with computers: An overview of findings. *Journal of Computer Assisted Learning, 10*, 202-215.

Hoyles, C., Healy, L., & Sutherland, R. (1991). Patterns of discussion between pupil pairs in computer and non-computer environments. *Journal of Computer Assisted Learning, 7*, 210-228.

Kafai, Y. B. (1993). Minds in play: Computer game design as a context for children's learning. Unpublished Doctoral dissertation, Harvard University.

King, A. (1989). Verbal interaction and problem-solving within computer-assisted cooperative learning groups. *Journal of Educational Computing Research, 5*, 1-15.

Kynigos, C. (1992). The turtle metaphor as a tool for children's geometry. In C. Hoyles & R. Noss (Ed.), *Learning mathematics and Logo* (pp. 97-126). Cambridge, MA: The MIT Press.

Lehrer, R., Guckenberg, T., & Lee, O. (1988). Comparative study of the cognitive consequences of inquiry-based Logo instruction. *Journal of Educational Psychology, 80*, 543-553.

Lehrer, R., Harckham, L. D., Archer, P., & Pruzek, R. M. (1986). Microcomputer-based instruction in special education. *Journal of Educational Computing Research, 2*, 337-355.

Littlefield, J., Delclos, V. R., Lever, S., Clayton, K. N., Bransford, J. D., & Franks, J. J. (1988). Learning Logo: Method of teaching, transfer of general skills, and attitudes toward school and computers. In R. Mayer (Ed.), *Teaching and learning computer programming: Multiple research perspectives* (pp. 111-135). Hillsdale, NJ: Erlbaum.

Liu, M. (1997). The effects of HyperCard programming on teacher education students' problem-solving ability and computer anxiety. *Journal of Research on Computing in Education, 29*, 248-262.

McCoy, L. P. (1996). Computer-based mathematics learning. *Journal of Research on Computing in Education, 28*, 438-460.

Mehan, H. (1989). Microcomputers in classrooms: Educational technology or social practice? *Anthropology & Education Quarterly, 20*, 4-22.

Miller, R. B., Kelly, G. N., & Kelly, J. T. (1988). Effects of Logo computer programming experience on problem solving and spatial relations ability. *Contemporary Educational Psychology, 13*, 348-357.

Mitterer, J., & Rose-Drasnor, L. (1986). LOGO and the transfer of problem solving: An empirical test. *The Alberta Journal of Educational Research, 32*, 176-194.

Noss, R. (1986). Constructing a conceptual framework for elementary algebra through Logo programming. *Educational Studies in Mathematics, 17*, 335-357.

Noss, R. (1987). Children's learning of geometrical concepts through Logo. *Journal for Research in Mathematics Education, 18*, 343-362.

Papert, S. (1980). *Mindstorms: Children, computers, and powerful ideas.* New York: Basic Books.

Papert, S. (1987). Computer criticism vs. technocentric thinking. *Educational Researcher, 16*, 22-30.

Papert, S. (1993). *The children's machine. Rethinking school in the age of the computer.* New York: Basic Books.

Papert, S. (1996). An exploration in the space of mathematics educations. *International Journal of Computers for Mathematical Learning, 1*, 95-123.

Reed, W. M., Palumbo, D. B., & Stolar, A. L. (1988). The comparative effects of BASIC and Logo instruction on problem-solving skills. *Computers in the Schools, 4*, 105-118.

Rieber, L. P. (1987, February). Logo and its promise: A research report. *Educational Technology*, 12-16.

Roblyer, M. D., Castine, W. H., & King, F. J. (1988). *Assessing the impact of computer-based instruction: A review of recent research.* New York: Haworth Press.

Simmons, M., & Cope, P. (1990). Fragile knowledge of angle in turtle geometry. *Educational Studies in Mathematics, 21*, 375-382.

Wiburg, K. M. (1989). Does programming deserve a place in the school curriculum? *The Computing Teacher, 17*(2), 8-11.

Wright, J. L. (1994). Listen to the children: Observing young children's discoveries with the microcomputer. In J. L. Wright & D. D. Shade (Ed.), *Young children: Active learners in a technological age* (pp. 3-17). Washington, DC: National Association for the Education of Young Children.

Yelland, N. (1995). Mindstorms or a storm in a teacup? *A review of research with Logo. International Journal of Mathematics Education, Science, and Technology, 26*(6), 853-869.

Time to prepare this material was partially provided by a National Science Foundation Research Grant, "An Investigation of the Development of Elementary Children's Geometric Thinking in Computer and Noncomputer Environments," NSF MDR-8954664. 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.

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

Logo News
Logo News
Logo News

Pack your bags for Logosium '99!

Logo Exchange will once again sponsor Logosium, the full day celebration of Logo learning, programming and teaching, as a pre-conference workshop at NECC '99 in Atlantic City. Logosium will be held Monday, June 21. A hearty lunch and terrific social dinner will cap off a day's worth of hands-on workshops, demonstrations, and discussions. For more information, as it becomes available, send e-mail to logosium@stager.org.

Please consider sharing your expertise with us as a workshop leader or presenter! Educators interested in presenting at Logosium '99 should send e-mail to logosiumpresenters@stager.org.

Articles needed!

The next *Logo Exchange* issue, due in April will focus on mathematics, beauty and Logo. The last issue of the year, due in early June, will feature great examples of student projects and exciting classroom project ideas. Please consider sending your favorite Logo project idea or examples of student work to logoexchange@moon.pepperdine.edu.

MicroWorlds Pro for Windows

By the time you read this, LCSI should be shipping MicroWorlds Pro for Windows 95-98 computers. MicroWorlds Pro's new features include a more sophisticated interface, powerful Web authoring tools, and the ability to share data with Excel. See **http://www.microworlds.com** for more information.

Boxer available free for the Mac

The long-awaited programming environment, Boxer, developed by Andy diSessa and Co. over many years at UC Berkeley is available for free download from **http://www.soe.berkeley.edu/boxer**. Aren't free versions of StarLogo and Boxer enough reason to buy one of those cool new Macintoshes? Both MicroWorlds Pro and Boxer will have articles about them in the next issue of *Logo Exchange*.

Hacking the programmable brick

All sorts of folks are working overtime to produce decent software for LEGO's RCX programmable brick.

Richard Maynard has created a version of UCBLogo for the PC designed to control the brick. Download it at **http://www.netway.com/~rmaynard/**.

Those of you interested in running the commercial LEGO Mindstorms software on a Macintosh might be interested in the patch available at **http://www.enteract.com/~dbaum/lego/nqc/mac.html**. This patch allows the PC based Mindstorms software to run under Virtual PC on a Mac.

Polygons and More

by WILLIAM J. SPEZESKI

As a Logo practitioner, I am always looking for interesting ideas to build a project around. One of my recent thoughts was to use Logo to design buttons that could be used as clickable links for Web pages. The tool that I felt would be very useful was a procedure to draw a square with rounded corners. More generally, it would be valuable to have a procedure that would draw different polygons with rounded vertices. And so I begin the article in search of a procedure to do just that.

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 slight alteration

```
to polyr :n :l
   repeat :n [ fd :l arcr 15 360 / :n ]
end
```

is a procedure that produces something close to the desired result, rounded vertices. The RT command in POLY is replaced with ARCR (arc right), the usual procedure that draws an arc in a clockwise direction. Here the radius of the arc is fixed at 15 and the span of the arc is identical to the exterior angle needed to draw an N-sided polygon.

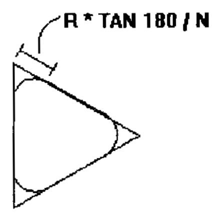
Procedure POLYR is quite usable, but it has a few deficiencies:

1. The radius of the arc is fixed.
2. The size of the figure is not easy to control.
3. The starting position is no longer at a vertex.

After a using a little trigonometry, the procedure can be rewritten to do exactly what we want. A revised version of procedure POLYR is shown below:

```
to poly.round :n :l :r
    local "adj
    make "adj :r * tan 180 / :n
    pu fd :adj pd
    repeat :n [ fd :l - 2 * :adj arcr :r 360 / :n]
    pu bk :adj pd
end
```

Don't panic! This is not as bad as it looks. Here is a summary of what this procedure POLY.ROUND will do:



1. Its third parameter *r* controls the size of the arc radius.
2. It produces a polygon with rounded vertices whose line segments coincide with portions of the enveloping regular polygon drawn by POLY with the same values of *n* and *l*. (Its size is controllable.)
3. The starting (and stopping) position of the turtle is at a vertex of the corresponding enveloping regular polygon drawn by POLY.

In order to round the vertices of a polygon and preserve the polygon's size, each side of the polygon must be shortened (adjusted) a little bit at each end. That "little bit" (ADJ) is equal to R * TAN 180 / N in our case. Here R is the radius of the arc and TAN is the trigonometric function *tangent*. You may have to substitute SIN x/COS x if the version of Logo that you are using does not have the TAN function available. Creating round vertices with procedure POLYR also shifts the starting position of the turtle. Compensating adjustments are made in procedure POLY.ROUND to position the turtle initally at a (virtual) vertex with the initial FD and closing BK commands.

Coming back full circle to Web page button design, procedure POLY.ROUND is a very useful drawing tool that will produce a scalable polygonal button shape. But so much for

Web buttons. Designing Web buttons is truly a very interesting Logo project. Having said that, however, I will leave that discussion to another time. What is more interesting at the moment is exploring procedure POLY.ROUND.

Procedure POLY.ROUND

Yogi Berra is famous for his quote "If you come to a fork in the road, take it!" Web page button design was the impetus for developing procedure POLY.ROUND. But now we will leave the world of Web page buttons and take the fork in the road, digressing with a brief discussion of the procedure that was developed above.

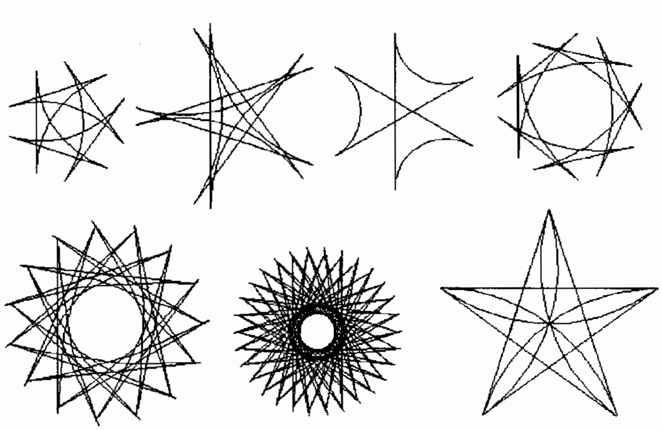
Procedure POLY.ROUND will draw polygons with rounded corners as long as the radius of the arc *r* remains a reasonable length. Looking at the expression

```
L  -  2 * R * tan  180/N
```

in the procedure, it is clear that for appropriate large values of R, this difference will take on negative values. When we get negative values of this expression, however, the fun begins! Going FORWARD a negative amount results in moving the turtle BACK, while going BACK a negative amount results in moving the turtle FORWARD. In fact, for larger values of R, procedure POLY.ROUND behaves a lot like

```
to polyb :n :r
   repeat :n [ bk :r arcr :r 360/:n ]
end
```

The above procedure is similar to procedure POLYR with FD replaced by BK and parameter *l* replaced by *r*. Both POLY.B and POLY.ROUND (with R>L/2*TAN 180/N) will produce some very dramatic effects. Let's look at a few examples.



The figures were generated with the following superprocedure:

```
to showit
   cs fs
   set -320  160  0 polyb        5 100
   set -150  100  0 poly.round  5  30 145
   set   30  110  0 poly.round  3  20  50
   set  150  100  0 poly.round  7  50 155
   set -300 -100  0 poly.round 15  20 450
   set  -70 -100  0 poly.round 30   5 750
   set  140 -100 18 poly.round  5  50 180
end
```

It uses procedure SET below to arrange the objects on the screen:

```
to set
   pu setxy list :x :y pd
   seth :z
end
```

Editor's Notes

The Logo programs in this article were written to work in Terrapin dialects of Logo. If you are using an LCSI dialect, like MicroWorlds, make the following changes.

- You can eliminate the `local "adj` line in `poly.round`.
- Change the SET procedure to SETUP. Set is a primitive in MicroWorlds.
- Change `cs fs` to CG in the SHOWIT procedure.

Conclusion

This article discusses an interesting procedure POLY.ROUND that draws regular polygons with rounded vertices and has some interesting side effects. But more importantly, it hopefully illustrates more of the hidden surprises and sense of discovery that Logo has provided over the years. Seymour Papert's gift to all of Logo is truly a gift that keeps giving. Thank you, Dr. Papert.

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
wspezeski@mcla.mass.edu

Papert, Logo, and Loving to Learn

by PAM GIBBONS

Mindstorms—where it all began . . .

In preparing to write this paper, I went to the bookshelf and retrieved my yellowed, dog-eared, now almost loose-leaf copy of *Mindstorms* (Papert, 1980). It had been quite awhile since I returned to the book that started it all for me, so I read it again. Perhaps mirroring what Papert described as how "we all forget about our thinking as children" (1980:41), I was surprised to read in those pages so many ideas, hopes and dreams about Logo and learning that I had long since internalised and accepted as part of some Logo community collective consciousness.

At first, I greatly admired Papert's writing probably because I agreed with what he was saying. It seemed so hard to make other educators understand what could happen if you put children and Logo together in the right environment—the things I was observing on a daily basis—that it was very exciting to find that a respected researcher was actually writing about the same things. I was fascinated by the theoretical discussion underlying Papert's observations. I was inspired by his propositions of educational revolution and what the future of education could hold. I entered the Logo community and conducted my own research, inspired by Papert's notions of such things as objects to think with, thinking about thinking, computational metaphors, sub-procedures and mind-sized bites, and the wonder of recursion. Most gratifying, however, was Papert's ready and emphatic recognition of the cognitive-social universe of programming and the quality of human relationships in Logo environments.

While the Papertian roots of many of these themes may have faded in my memory, one of Papert's observations has stayed with me clearly and strongly—that which he describes as the mathophobia (fear of learning) endemic in our society. In an alarming recognition of what once was a way of being for all of us, Papert observed that "Children begin their lives as eager and competent learners" (1980:40). I found this observation alarming because, particularly in schools, it was clear that most children had lost this eagerness and competence very early on in their formal education. Papert (1980:42) noted that:

> "An unknown but certainly significant proportion of the population has almost completely given up on learning. These people seldom, if ever, engage in learning and see themselves as neither competent at it nor likely to enjoy it."

The fact that this phenomenon was widespread but not absolute suggested, however, that it was not inevitable, and if not inevitable, then perhaps conquerable. Stated as a truism, Papert's comment (1980:42) that

> "The extent to which adults in our society have lost the child's positive stance toward learning varies from individual to individual"

begged for serious investigation. Why was this so? What can educators do about it?

Why doesn't everyone love to learn?

Years later, my doctoral research focused on individual differences, cognition, and recursion in Logo programming—truly Papert inspired themes (Gibbons, 1993; 1995). Based on the neuro-psychological cognitive theories of Luria (1973) regarding individual differences, and Activity Theory (Leont'ev, 1981; Vygotsky, 1978) regarding the plasticity and cultural situation of cognitive abilities, I explored the ways in which individuals used and conceptualised recursion in Logo programming. Two types of individuals' dominant cognitive styles were identified and studied—for ease of discussion in this paper, these are characterised as either simultaneous (relational) thinking or successive (procedural) thinking.

From this research I discovered that simultaneous learners demonstrated an eagerness and curiosity for new knowledge; they placed high value on "understanding" concepts rather than on memorisation—their activity was characterised by the pursuit of knowledge for its own sake. In marked contrast, the successive learners demonstrated resistance to receiving new instruction and reluctance to adopt new techniques or modify old ones. This discrepancy was reflected in the pairs' widely divergent perceptions of what constituted knowledge and learning: So polarised were these perceptions that widely divergent educational outcomes between these cognitive processing types were inevitable.

The identification of the existence of a polarity of beliefs between individuals regarding the very nature of learning itself, was entirely consistent with diSessa's (1985) observations of contrasting "intuitive epistemologies" held by two case study subjects. DiSessa observed that one student maintained a view of "understanding physics" that was consistent with diSessa's own p-prims epistemology (1983). In contrast, the other student was perceived to hold the view that knowledge of physics resided in its equations and formalisms (with intuitive knowledge only serving to confound understanding). DiSessa noted the resistance of the latter student to adopting a more systemic approach and forwarded the expectation that this student would be at a distinct disadvantage in learning physics. The findings of my own research were consistent with diSessa's observations and suggested a theoretical foundation for his and Papert's propositions.

In my research, successive learners demonstrated the assimilation of knowledge by remembering sequences of processes or instructions for individual instances of similar cases. For these subjects, learning is embodied in the memorisation of unrelated pieces of information or ritualistic procedures. The notion of education thus becomes onerous and cumulative; the presentation of new information involves the extension of the long list of things to be remembered rather than supporting existing knowledge through the development of a more robust mental model. For the simultaneous learners, however, new learning experiences were welcomed in their promise of providing further clues in the construction of uniform and universal mental models. New information is, for the latter type of learner, thus linked to the long-term reduction of mental activity (at the expense of short-term intellectual effort) and, in a programming context, to the ultimate ability to implement efficient and elegant procedures.

How can a love of learning be encouraged?

Traditional Western learning environments have been marked by an emphasis on activities supported by successive processing strategies. Although using different terminology, this observation was also noted by Papert in *Mindstorms* (1980): the mnestic outcomes of tasks favouring successive processing are easily defined and measured with educators thus being more readily accountable for their work. This is in contrast to the problematic nature of assessing the development of thinking or problem solving skills associated with simultaneous information processing. If schooling almost exclusively encourages successive abilities through the nature of traditional task demands then successive abilities will progressively improve by default and without deliberate intervention at the expense of simultaneous abilities.

The concern is that we may then be "breeding" learners who dread learning when we should be actively pursuing the type of tasks that will invoke, and thus develop, simultaneous abilities. In this way, educators may deliberately promote simultaneous processing abilities that in my research were associated with learners who enjoyed learning. Efforts to encourage the development of simultaneous abilities should not, however, be confused with Papert's (1985) ideological concerns regarding the "forcing" of style. Rather, it is proposed that attention should be given to the development of both simultaneous and successive information processing modes so that learners might be better placed to manifest a style preference instead of, by default, an inappropriate style necessity.

Full circle

Traditional school environments easily create environments in which successive thinking reigns supreme—the time to design learning environments to cultivate simultaneous thinking is long overdue. So, what would such learning environments look like? Learners will be engaged in tasks that demand relational thinking, that involve keeping a number of items in mind at once, that demand the identification of non-sequential relationships between components, that require planning and contemplation. Learners will concentrate more on solving problems than on knowing facts (resonance with Brian Harvey's comments regarding the Internet (1997). There will be discovery; there will be creativity, there will be a richness of human relationships; learners will have time to play and to think; variety will be valued as an educational outcome; educators will not be driven by time limits, standardised test scores or expediency. Learners will love learning.

It sounds like *Mindstorms*. It sounds like Papert. I have conducted my Papert-inspired research and have analysed its findings, and yet it simply confirms Papert's 1980 call to arms. Papert's call may have ended up being more of an evolution than a revolution and the technology may have marched on, but his ideals have stood the test of time.

See LOVING TO LEARN (Page 36)

THE LAST WORD: COMMENTARY

Explaining Yourself

by BERNARD NEWSOME

Sitting here at my Power Mac G3 266, struggling to adjust the color in PhotoShop in order to print out a red rather than a brown image of a small French cake, Le Melodie. I marvel at what a distance computers and their programs have come since I was at University in the Fifties. That machine, which seemed only to deal with exam correlations, or the playing of Nim, kept two large rooms humming to the gentle sound of many valves. Possibly before I die, every man, woman, and child will have on their kitchen tables a vast improvement on what I have today, and breakfast will consist of multiple connections to the Internet instead of rushing out to the office or school. The American predilection for fast foods, though, is likely to increase.

Like most of my generation, I just use the power of the computer to do what I want to do. Most of that is limited, and my wife is even more hesitant than I am at learning a new set of procedures. It was only just yesterday that I clicked on Internet Explorer, and did a bit of surfing for the first time. A couple of my mates said I could with benefit clock onto The Connected Family's Seymour Papert Web site (**www.connectedfamily.com**), and revisit *Mindstorms* in an updated. I did that, slid into the site of master and disciples, and read the lecture, Child Power: Keys to the New Learning of the Digital Century.

Much there to admire, I thought. (I thought the Internet had little of interest in it until recently.) Empowering children because the computer gives them, through their extraordinary method of playway learning, an advantage over goops like me.

> Could their wisdom added together give us pedagogy and a program for educating for the future—a theoretical base operating on, and being informed by, a practical understanding?

Radicalizing and subverting traditional learning establishments, called schools, is possible when multiple ages and multiple activities go along together. A social setting of relative experts or relative beginners, offers a context for online help of the human kind. And then the use of the computer for constructing, rather than transmitting. Yes, I remember being taught MicroWorlds and turtle graphics programming by a tidily 9-year-old. Goodie, Goodie.

But I can't see schools giving up their position in the educational system, nor governments allowing it to happen. Why? One particular reason is we assume that if a child is not successful in a traditional school it is the child who has failed, whereas if the child is unsuccessful in an alternative school, or in an alternative program, then it is the school or the program that has failed. The corollary of this, of course, is that if the child is successful in the first place, it is to the school's credit. But, if successful in the second, he/she was clever enough to make up the leeway.

America has had some outstanding theorists and practitioners in education, I mused. Dewey was a profound initiator; in modern times Bruner, and in even more recent times, Donald Graves. And of course Papert himself. Running sideways across that tradition is George Kelly, the psychologist of personal construct theory, and then sideways to the Russian trio of Vygotski, Bachtin and Luria, sideways to Piaget, and head on to Polanyi. (I know you have your favorites too!) Could their wisdom added together give us pedagogy and a program for educating for the future—a theoretical base operating on, and being informed by, a practical understanding?

When Dewey advocated learning through experience, he promoted a problem-solving environment. Every new personal or social venture, in whatever educational context, requires adjustment of previously internalized routines for dealing with what we find. These internalized routines, some call them skills, are all very well whilst the earth stands still, or whilst all experience is predictable. But, as we all know, if we could predict accurately which horse will win the next race, or which poker machine, operated by a program, is set for the jackpot on the next turn of the wheel, we'd have no need to learn anything new; what's more, we'd all be on holiday in the Bahamas.

Part of what we do requires little adjustment. We have learnt the routines; we carry them out, and are satisfied. (I know how to make the bread I like.)

But the future, at all levels, is not known. Thus our tacit knowledge, which Polanyi clearly saw as the crucial mechanism for understanding what confronts us, is continually in need of more than recursive resetting; the storehouse needs to be expanded to cope with the ever present newness of the changing circumstance. Equally, though, our tacit knowledge is not to be abandoned, for what we have is the fruit of past encounters. Unless what we find is totally unknown, we have enough to dig into the slippery slope and stabilize, although to climb upward may require expansion of the system.

Extrapolating to curriculum content and structure, the learner in any area brings to the task a body of expectations and a repertoire of routines which have worked in the past, repertoires both personal (Kelly) and social (any post-'70s guru). Likely, today's encounter will be familiar. But when it is not, new procedures have to be learnt and incorporated before a satisfactory outcome can be achieved. Curriculum content thus should be not totally new, and it should be structurally related to what went before. Build on the old so that the learner doesn't drown.

On another tack, Bruner, in *Man A Course of Study*, made the audacious claim that curriculum content of itself does not matter—study the behaviors of Salmon and Eskimo and you will find similar conceptual schemes applied to make sense of the data. Indeed, he argues that the same concept can be learnt by children at all ages, albeit at different levels of complexity and inter-relatedness. If that is right, it would suggest that vertical sequencing in curriculum makes sense when content provides richer pastures for exploration of already familiar conceptual structures.

But enough of the big boys. What about those three Xs I found on the Mamamedia site (**www. mamamedia. com**) developed by Papert disciple Idit Harel into a practical program of multimedia learning: EXploring, EXpressing and EXchanging? All the theories grounded in children's learning activities. I'd like to add one more, Explaining. The How of the doing, or the question of learner to knower, How do I do that?, promotes the acquisition of skills. The Why for the doing, or the question of learner to learner, Why are you doing that?, or Why do it at all?, develops the higher order understanding of the relationship between things; makes the practitioner into the theorist. The problem for most schools is that the control of the why is with those who can do, and know why they do, which has tended to be seen as the teacher.

Papert's work suggests a better route.

About the author

Bernard Newsome is a retired professor of language and literature. He is the author of *Understanding Children Writing and Talking, Writing and Learning*. He makes Shaker furniture and is artist's assistant to his wife, Mary, who is a printmaker, painter, and photographer. Since retirement, he has become an enthusiastic computer user and Logo hacker. Bernard lives in Melbourne, Australia.

Bernard Newsome
newsholm@bigpond.com

LOVING TO LEARN / Cont. from Page 34

As educators, we can pay no greater homage than to continue to pursue the fulfilment of these ideals. LX

References

diSessa A. (1983). Phenomenology and the evolution of intuition. In D. Gentner and A.L. Stevens (Eds), *Mental models*. 15-34. Hillsdale, NJ: Lawrence Erlbaum Associates.

diSessa A. (1985). Learning about knowing. In E. Klein (Ed.), *Children and computers: New directions for child development* (28). 97-124. San Francisco: Jossey Bass.

Gibbons, P. (1993). Recursion: An analysis of individual differences in Logo programming. Unpublished doctoral thesis. University of Sydney.

Gibbons, P. (1995). A cognitive processing account of individual differences in novice Logo programmers' conceptualisation and use of recursion. *Journal of Educational Computing Research 13*(3), 211-226.

Harvey, B. (1997). Stop the internet, I want to get off. *Logo Exchange 16*(1), 18-19.

Leont'ev, A. (1981). *Problems of the development of the mind*. Moscow: Progress.

Luria, A. R. (1973). *The working brain: An introduction to neuropsychology*. Middlesex: Penguin.

Papert, S. (1980). *Mindstorms: Children, computers and powerful ideas*. New York: Basic Books.

Papert, S. (1985). Computer criticism versus technocentric thinking. Logo85: Theoretical Papers. Cambridge, MA: Massachusetts Institute of Technology.

Vygotsky, L.S. (1978). *Mind in Society*. Cambridge MA: Harvard University Press.

About the author

Dr. Pam Gibbons teaches at the Australian Catholic University-Mount Saint Mary Campus, 179 Albert Rd., Strathfield. NSW. 2135 Australia.

Pam Gibbons
p-gibbons@mary.acu.edu.au

Fat Crayon Multimedia

Using Kid Pix

Turn your primary classroom into a multimedia studio and watch the lights go on, the images roll and the action begin. Any anxiety about using multimedia will evaporate as this imaginative guide inspires your students to explore and express ideas and concepts. Twenty lesson plans and seven slideshow presentations teach language arts and math using Kid Pix and Kid Pix SlideShow—child-friendly programs with the same tools as "grown up" multimedia software. Along the way, you and your students will master multimedia basics: computer drawing and painting, inputting text, recording sound, importing graphic images and simple animation and video. Every 10- to 15-minute lesson was designed in a real classroom and includes an outline of its curricular basis and the multimedia and computing skills involved. Illustrated step-by-step "Cheat Sheets" in each lesson and a generous Teacher Help section guarantee multimedia explorations that are quick, easy and productive.

Marybeth Kampman
Member Price $22.45 (Nonmembers $24.95)
140 pages, ISBN 1-56484-138-3, ©ISTE, 1998

Lesson plans and ideas use Kid Pix 2 and Kid Pix SlideShow for Mac; may be used with other versions of Kid Pix with a few modifications (software not included)
AUDIENCE: Grades K–2 teachers & students

International Society for Technology in Education
iste
800.336.5191 *(U.S. & Canada)*
541.302.3777 *(International)*
www.iste.org

To order call 800.336.5191 or visit our Online Bookstore at www.iste.org

Beyond Generation X...

GENERATION WHY

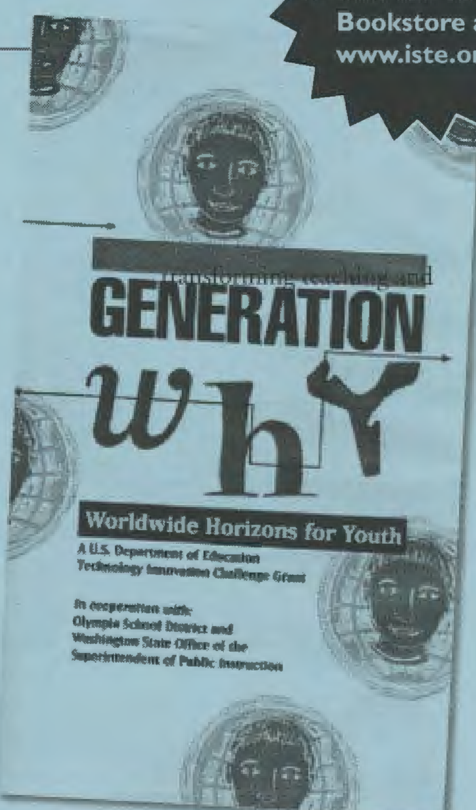
Generation WHY Video

Beyond Generation X is Generation WHY, the bold and original idea that's learning by extensively involving students as education's partners and leaders.

No longer do teachers need to spend long hours learning technology applications on their own. The Gen WHY course teaches students these applications as well as the skills needed to partner with teachers to improve student learning through infusion of technology.

The video provides an introduction to all phases of the model, presents the 12 curriculum units taught in the Gen WHY class and explains how students work with teachers to implement technology-infused lessons.

60-minute VHS Video, ISBN 1-56484-141-3, ©ISTE, 1998



International Society for Technology in Education
iste
800.336.5191 *(U.S. & Canada)*
541.302.3777 *(International)*
www.iste.org

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

Non-Profit
Organization
US Postage
PAID
ISTE