DOCUMENT RESUME

ED 257 452 IR 011 685

AUTHOR Hawkins, Jan
TITLE The Interpretation of Logo in Practice. Technical Report No. 34.
INSTITUTION Bank Street Coll. of Education, New York, NY. Center for Children and Technology.
SPONSORING AGENCY Spencer Foundation, Chicago, Ill.
PUB DATE Mar 85
NOTE 38p.
PUB TYPE Reports - Research/Technical (143)

EDRS PRICE MF01 Plus Postage. PC Not Available from EDRS.
DESCRIPTORS *Classroom Techniques; *Cognitive Development; Development; Elementary Education; *Elementary School Students; Individual Differences; Instructional Innovation; Interviews; Pilot Projects; Problem Solving; Programing; Programing Languages; Research Methodology; *Social Development; *Teacher Attitudes; Teacher Role; *Teaching Styles
IDENTIFIERS *LOGO Programing Language

ABSTRACT

Designed to help understand the cognitive and social effects of children's classroom experiences with LOGO and computers, this study presents an account of the ways in which two elementary school teachers thought about, grappled with, and practiced LOGO in their classrooms over a 2-year period. The account is organized chronologically, first describing the pilot period when microcomputers were placed in the classrooms and LOGO became part of classroom activities for 2 months before the school year's end; then the experiment's first year, including a recounting of the teacher's expectations, classroom work which shifted focus from LOGO as a learning environment for general problem solving skills to LOGO as a context for learning about programming and computers, and reasons for the shift; and finally, the second year, when teachers developed instructional strategies and revised classroom work to provide a particular type of structure to LOGO learning. Subjects were 25 third and fourth graders (11 boys, 14 girls) and 25 fifth and sixth graders (11 boys, 14 girls) encompassing a variety of ethnic and socioeconomic backgrounds and a range of achievement levels. (MBR)

CENTER FOR CHILDREN AND TECHNOLOGY

The Interpretation of Logo in Practice

Jan Hawkins

Technical Report No. 34

Bank Street College of Education 610 West 112th Street NY NY 10025

Abstract

Innovations in education often make their way into classrooms through complex processes that allow them to be interpreted in relation to the existing learning setting. This paper analyzes the ways in which two upper-elementary teachers interpreted and reinterpreted the value of the computer programming language Logo, and the ways in which it should be taught, over the course of a two-year experiment.

The Interpretation of Logo in Practice

Jan Hawkins

Technical Report No. 34

March 1985

THE INTERPRETATION OF LOGO IN PRACTICE*

Jan Hawkins

Innovations in education seldom simply arrive and radically redefine the activity or meaning of learning situations for participants. Rather, educational innovations insinuate themselves gradually, affecting certain aspects of the setting, and are interpreted and shaped by the participants. As one aspect of a research program conducted at the Center for Children and Technology, we were concerned with understanding how a particularly promising educational innovation--using computer technology to learn general problem-solving skills through programming in the Logo language--was assimilated into the classroom.

Often embodied in radically new materials (e.g., new math), new perspectives on learning (e.g., open education), or new technologies (e.g., ALM accompanied by language labs, television, and computers), educational innovations have complex histories in individual classrooms and in American education in general. Like ideas or text materials, such innovations are interpreted in terms of the knowledge, experience, and setting of the teachers and students who encounter them. Thus, they do not change the broad educational landscape by leveling what was already there, but instead offer opportunities for reshaping the existing content in individual settings. Similarly, one must expect interpretation of the innovations themselves as they are practiced in the complex systems of classrooms. This is what happened with Logo.

The following is an account, itself an interpretation, of the ways in which two teachers thought about, grappled with, and practiced Logo in their classrooms over the course of two years. One source of information is utilized: the perspectives of the teachers as expressed in interviews throughout the two years, and in journals they kept during the first three months of the experiment. The coherence of the teachers' developing views--based on what they thought, tried, and observed--is important to preserve as an interpretive voice for the experiment. The teachers began with a set of beliefs and expecta-

---

*This research was supported by a grant from the Spencer Foundation.

tions for Logo which were revised and developed in ways that could be systematically described in the course of their attempts to fit their classrooms to Logo, and Logo to their classrooms.

This is but one of several accounts that can be given of the interpretation of Logo in this experiment. The research program was designed to help us understand the cognitive and social effects of children's experiences with Logo and computers in classrooms. In this effort, we have conducted many studies and collected several types of information, the findings from one study informing the methods to be used in the next. The work of the Center has included: longitudinal case studies of individual learners; experimental studies of planning skills and understanding of programming concepts; assessments of expertise; observational studies of interactions in classrooms; videotaped studies of peer collaboration; and analyses of collaborative skills in group-structured situations. It was essential that all these studies--careful, detailed fragments of the whole--be seen in a single context, that of the classroom experience of Logo. In order to provide this descriptive context, teachers were interviewed, classrooms were regularly observed, lessons were recorded, and records of children's work were collected. The interviews with the teachers were particularly rich and, taken together, provide a coherent picture of what took place.

The Development of Logo

Logo was developed as a programming language for children by a team at MIT in the early 1970s; Seymour Papert (1980) has become the principal spokesman. The language was designed to introduce children to programming concepts and, through this experience, to develop powerful higher order thinking skills that could be transferred to other contexts. Logo is accompanied by a particular pedagogy: children are to learn through self-guided discovery methods, pursuing their own goals and ideas with minimal adult intervention or systematic presentation of concepts or skills. Thus, it has been claimed that children develop general problem-solving skills through self-initiated and self-guided exploration of Logo. Logo has been described as a special and rich environment for the acquisition of high-level logical and reasoning skills.

The availability of Logo also filled a need in the educational community: computers were rapidly being acquired by schools, but without good software or clear notions for defining their role and use. Logo offered a powerful way of using the machines, a rationale with broad-ranging and highly desirable learning outcomes, accompanied by respect for the child's natural capacities and initiative. Our research program was designed to look closely at the development of children's

understanding in the most salient educational environment offered by our culture--the group-learning setting of classrooms.

The Teachers and Classrooms

The study was located in two classrooms at a private school in Manhattan. One classroom included 25 (11 boys, 14 girls) 8- and 9-year-old children (third and fourth graders); the other consisted of 25 (11 boys, 14 girls) 11- and 12-year-old children (fifth and sixth graders). The children encompassed a variety of ethnic and socioeconomic backgrounds and a range of achievement levels. Each classroom had six microcomputers during the 1981-1982 school years. Both the younger and older groups had three Apple II Plus computers and three Texas Instruments (TI) 99/4 computers.

These classrooms were selected because the children met our age requirements, and because the teachers expressed interest in participating in the research. These teachers (two men), as well as the school math coordinator (a woman), attended a six-week summer course in Logo conducted by the New York Academy of Sciences, with Dr. Papert as one of the instructors. Skeptical at first, both teachers became engrossed in the topic of programming and completed the course feeling cautiously optimistic about Logo's potential as an important learning experience for children. The teachers and the coordinator continued to meet regularly to advance their understanding of Logo and to help one another with their programming work during the first part of the school year.

The participants were experienced and talented teachers. The basic philosophy advocated by the Logo developers was compatible with the teachers' perspectives on education, generally described as a child-centered learning approach.

Over the course of the experiment, the teachers acquired considerable skill in Logo in particular, and programming in general. In addition to the two years of work with the children in their classrooms, the teachers developed and taught courses in Logo and computers in education to master's degree candidates in a graduate program. One of the teachers wrote a book about Logo applications in education, and guides for educational software designed to introduce programming concepts.

In the spring of the pilot year, six microcomputers were placed in each of the two classrooms, and Logo became a part of the classroom activities for a two-month period--until the end of the school year. Throughout the next two school years, the teachers sought to engage the children's interest in and develop their understanding of Logo.

Our research period, therefore, covers three groups of about 50 children each: one two-month pilot period, and two complete cycles of school years. Thus, the teachers had the opportunity to work out and rework their approach to the innovation, modifying the introductory material and course of instruction for each successive group.

The Evidence

In addition to frequent informal discussions, each teacher was formally interviewed ten times throughout the research period. These interviews constitute the bulk of the material discussed here. Each session lasted between 45 minutes and two hours, and was conducted by a member of the research team with whom the teachers were familiar. The interviews were structured, covering some of the same questions each time and focusing on new issues as they arose. The teachers were invited to reflect and speculate on their experience, and the interviewer encouraged them to develop their ideas throughout the discussion. In addition to these dialogues, the teachers kept journals of their experiences with computers during the initial two-month pilot period. This material is cited where appropriate, particularly because it provides some detailed information about what the teachers initially thought about the innovation.

Method of Analysis

The interview sessions were tape-recorded and transcribed for analysis. Qualitative analyses were performed on the material. The transcriptions were read several times by several readers in order to develop a system for categorizing the material. The interviews were divided into segments according to class, date, and overall theme. The themes included: definitions and redefinitions of Logo as a classroom element; discussions of cognitive abilities and transfer; evaluations of Logo, and problems encountered; strategies for supporting learning or teaching Logo; and the social context of the classroom. The interview material was then reorganized by theme and date so that changes in the experience within each of these categories over the two years could be noted.

The Role of Research

Computers and Logo were introduced into these classrooms as part of a research project to document and analyze the cognitive and social effects of this technological innovation for education. A variety of studies were conducted throughout the two years, both in the classrooms and with individuals and pairs outside of the classroom context. Rather than imposing a point of view or curriculum, we, as researchers, were interested in how the teachers themselves chose to incorpo-

rate this innovation. It was a collaborative venture, and what each of us saw had some influence on what the others thought and did. The children were also aware that they were participating in an "experiment" which was intended to try out computers and Logo in classrooms. This knowledge undoubtedly influenced the ways in which they participated in the work.

The following account--a description of what the teachers did with the computers and how they thought about their value for the children--will be divided chronologically into two sections. First, we will describe the pilot period and the first year of the experiment for each classroom. The activity during the second year, which saw a significant reorganization, will be described in the second section.

Two general themes can be used to organize the teachers' accounts of this work. They implicitly struggled with two questions: "What is Logo and what is its value?" and "How should I organize a learning experience with Logo?" In their interviews, the teachers defined and redefined Logo from the perspective of those who organize the learning experience of novices. Thus, they grappled with understanding how Logo could be used as a learning tool in the classroom, and with defining its value for children. These definitions and evaluations developed over the course of the two years.

The fact that these interpreters' relationship to Logo was as "teacher" in a particular kind of setting is critical. The development of a perspective is based on the mode of engagement: the teacher's perspective will be quite different from the researcher's which, in turn, will differ from that of the software developer. These teachers were concerned with Logo as part of a complex, ongoing, and multi-faceted program of learning that they had carefully constructed during their years of teaching. Each of them went through three cycles of planning and revising this learning experience for Logo.

Section 1. Pilot Period and First Year: What Is Logo and What Is Its Value for Children?

The Teachers' Expectations

Before beginning the Logo experiment with their summer training sessions, both teachers (hereafter known as Dan [the younger class] and Jeff [the older class]) were somewhat critical of and skeptical about computers and their role in the larger culture. Both reported that they had had frustrating experiences with the technology. At the culmination of a law suit, Dan had been told that there was no legal basis for his complaint since he had been the victim of a "computer error": "It's in the computer. There's no responsibility."

Jeff, on the other hand, recognized that computers were part of our culture, but there was great potential for misuse. His concern was that use of the technology by children might be trivialized and focus on video games rather than educational applications. However, both teachers were willing to learn more about the technology and its potential for classroom use. Logo seemed a promising possibility for children in their classrooms because of its claim to engage and develop general problem-solving skills.

Both teachers found the summer seminars to be engrossing and challenging. One reported that he had had trouble picking up programming concepts, but that he very quickly became immersed. He described the learning process as one of reaching a plateau of knowledge, struggling for a while, and then reaching another plateau--a unique experience for him because he was used to effortless learning and acquiring skills without difficulty. Consequently, he sometimes felt frustrated and stupid as he wrestled with Logo. The other teacher found the logic of programming easy to learn, but reported that "it was difficult to orchestrate the commands for a purpose"; that is, it was not always easy to figure out how to construct a program to suit a desired goal.

At the end of the training period, both teachers felt that they had made significant progress, but that they had a considerable way to go before they could be fully functional with the language in the classroom setting. "It was more exhilarating than I thought; I was pleased when I understood things."

At the beginning of the experiment, Dan and Jeff were enthusiastic about the educational potential of Logo. They found the claims for the development of general problem-solving skills and the self-discovery pedagogy to be persuasive. Although they had some reservations, they expected that Logo could be a powerful, albeit limited, learning tool for children to explore. Two kinds of expectations were expressed. First, that Logo could be a unique type of school task for the practice and development of certain types of thinking skills:

> It could enhance kids' ability to deal with problems abstractly.
>
> * * * * *
>
> It could help them to structure their thoughts and think about things on an abstract level--maybe make it easier for them to plan out something on paper, communicate at a meeting, draw plans for something to build, understand geometry.

* * * * *

It might help kids doing logical and analytic thinking.

* * * * *

It's real work as opposed to assigned problems to solve. They'll learn to be better thinkers, to discuss questions and mull things over--a sense of self-initiation, setting up tasks and solving them. I hope they'll be able to discuss a process whereby they solve a problem better, how to go about it, and I hope it will generalize to any problem, like scenery for a play. They'll be able to break it down into pieces, to use analytic skills.

With respect to the types of skills engaged, the teachers felt that Logo might be a vehicle for (1) developing analytical skills that could be used to approach and solve many types of problems; and (2) giving kids conceptual tools and language for discussing, presenting, and communicating about the problem solving process.

However, each teacher also expressed reservations about Logo's possible limitations:

I'm not sure, though, to what extent working on computers makes kids deal with the most important questions they have to deal with--it's for intellectual, not affective or value questions. Analytic thinking could be helped by computers, but I'm not sure of its role in life--the breaking down of a situation into components.

* * * * *

I still sometimes wonder about its purpose--what really do children learn from it?

It is interesting to note that neither teacher discussed the value of Logo in terms of the importance of learning programming as a skill; it was interpreted primarily as a conduit to other general abilities.

The second major expectation was that Logo could be a learning environment which supported self-initiated, expressive learning. The innovation was valued for the claim that children could individually engage in developing their own skills, at their own pace, and in the context of self-selected goals:

> **The best scenario as far as I'm concerned is children** working on their own creations, not drill or abstractions, or presenting of concepts. I'd like them to feel, "I want to follow my own ideas."
>
> * * * * *
>
> You should do whatever you feel like, according to your own imagination. You get immediate results, rather than have to find out all the things necessary to deal with programming--there's a possibility that kids will be working on their own ideas independently.
>
> * * * * *
>
> In a classroom, it's a different experience. There are times when you shouldn't be answerable to anyone else, it's completely personal, no other input. It's a piece of work that is theirs and they will talk about.

In terms of the classroom curriculum, both teachers thought that Logo was most similar to creative writing because it offered children a medium for expressing their own ideas. However, Logo was unique in that the skills and ideas developed in this expressive context were logical and analytical. In other words, the presence of the computers and the selection of the software were not sufficient unto themselves; the mixture of cognitive content, medium, and pedagogical rround presented a new configuration for teachers to incorporate into their classrooms. Each element in the mixture was reinterpreted by the teachers as the experiment proceeded. Throughout the pilot period and first year, the teachers sought to understand the features of Logo as a tool for developing cognitive abilities and as a new environment for self-initiated learning.

The Pilot Period: Classroom Work

For the pilot period, Dan decided to impose little structure on Logo learning. He was committed to the position that Logo was a self-expressive medium for children, fully supportive of self-discovery learning:

> It's different from most of what goes on except for writing. They can use it in their own way. If it's a powerful tool, they can use it themselves.

The principal structure he felt he needed to provide was the scheduling of work (two one-hour periods a week for each child, plus

lunch and before school; organization of partners), and helping with the complexity of the relative geometry of the turtle movements ("eliminate the geometrical traps"). Dan decided to show the children the basic commands, and then "have them play with it a lot. But I won't give them ideas. I want to see their own ideas."

Over the course of the six weeks, Dan had only two meetings with the whole class, in which he demonstrated the capability of the "sprites" and introduced the concept of programming. All other teaching was done individually. Occasionally, Dan wrote commands or small programs on the blackboard for the children to copy and try out. Scheduling presented a problem for him--how to arrange for all kids to work on the computers regularly. But he also reported that the presence of the computers loosened up the classroom space and kept half the class occupied, enabling him to work intensively with small groups on other things.

Contrary to expectations, ideas had to be supplied to the children, and Dan handled this problem on an individual basis. The projects some children chose to do were disappointing. The children looked to their world of experience with computers to find projects; many of them decided to do videogames, and were disappointed by their inability to reproduce them.

Additional problems encountered by Dan were: many children didn't know how to "take the next step" in their projects; children found the angle inputs required by turtle geometry to be a problem; the idea of information storage in computers was a problem for many--they were puzzled by the question: If the program is stored on a disk and you recall it, where is the information?

Like Dan, at the outset of the experiment Jeff decided to provide the children with the basic commands, and to allow them to follow their own interests and pace as they worked with Logo. He decided to begin with turtle graphics and bring in sprites later in the term. The children were encouraged to develop their own goals, and Jeff supported their learning either individually or in small groups. No large-group meetings were held to teach computer concepts; when the children wanted to know something, they either asked Jeff or an experienced peer. Ideas spread through the class, and Jeff found that children shared programs freely with one another. He anticipated that the children would have trouble with certain concepts (e.g., variables and directionality), but expected them to master these in three to four weeks. The children were scheduled for two periods a week with the computer, and they could elect to do additional work before school or during lunch. However, computer work was contingent on having finished all other assignments.

The Pilot Period: Reflections

At the end of the two-month pilot period, both teachers were excited about the experiment, and had observed some children doing sophisticated and interesting things with Logo. Dan commented that "Some kids got involved with computers on an abstract level and thought about computer problems in a way they wouldn't think about other problems." However, both teachers were beginning to express reservations about Logo which they made explicit in several ways. Contrary to expectations, they had noticed substantial individual differences in children's interest in and learning of Logo. A few children became thoroughly engrossed in the activity, whereas others showed no interest or even hostility. Jeff said, "I wasn't able to figure out how to get the uninvolved kids involved." He was dissatisfied with many of the projects chosen by the children, such as the replication of videogames.

Concern was also expressed about the depth of the children's knowledge. The teachers felt that many children's level of engagement and contact with Logo was too low to permit them to make sense of what was going on.

> They simply seemed to accept the logic of the computer on its own grounds rather than struggling to understand it--"If that's the way you do it, that's the way you do it."
>
> * * * * *
>
> I thought they would be going home and deciding how to program things, but no. They seemed to work on programming only when they actually worked on the machines.
>
> * * * * *
>
> I was not surprised at the direction--games, but I was surprised at how little discussion there was other than "what should we do next."

As Dan became more familiar with the Logo language, it became less clear to him just what depth of understanding was desirable for the children:

> The kids do understand it because they can use it. I see that there's a great limitation to my understanding--I don't really know what it means to "make quotes" or "read character." It's a whole program that I just have to accept, that I have never bothered to find out just what goes on in

> that program. So I discovered that I wasn't giving kids the opportunity to use things that they didn't understand. But in the same way, I had the opportunity, so it's led me to present elements in a way that is functional to them. It's possible to go a lot farther this way.

Neither teacher saw evidence of the transfer of the cognitive abilities gained through programming to other areas. Initially, they focused on the Logo language as a potentially powerful environment for developing general thinking skills; after two months they were beginning to be skeptical of this claim.

> I didn't see much cognitive effect. I didn't see them talk about programming, and I didn't see the skills showing up anywhere else.

* * * * *

> Transfer? It's hard to say where it shows up. They weren't even thinking that way about the computers a lot of the time.

The First Year: Revisions for Classroom Work

At the beginning of year 1, both teachers decided that more structure was required if the children were to become competent users of Logo. At this point, they tended to see this structure as further individual support for the discovery and development of Logo project ideas, and greater adherence to a work schedule by all the children.

Dan gave the children little preparation other than to introduce the functions of drawing and animation. However, he did decide to give greater support to the discovery of project ideas. Since 17 of the children in his class had been in the pilot Logo group, Dan decided to have them help the inexperienced children learn basic concepts. This strategy "ended up hard" because, when the experienced children were paired with the inexperienced, "the experienced ones already had the idea of programming and took over."

Problems and techniques were written on the blackboard by Dan as a source of information. The problems were intended both to challenge the kids and to give them ideas about what could be done. He also supplied written information sources, so that the children could use them at the computers and, if they wished, take them home. Dan continued to work with the children individually or in small groups, as the need arose; there were no large-group sessions. He felt that it was too early to begin critical discussions about the technology:

> It would be interesting but they're very young, very young to have a critical perspective. I'm going to get a lot of these videogame junkies talking about how wonderful it is. So that puts me in the position of having to do editorializing for them, or having to sit there saying, uh huh.

Jeff also organized the first year's work around helping the children to find and work on their own projects. Teaching took place individually and in small groups, and Jeff encouraged the experienced children to help the inexperienced ones. Meetings were held to talk about the projects under construction, and about computers in the larger cultural context. Computer work was no longer contingent on finishing other work.

There was increasing recognition of the complexity of learning Logo. The teachers' comments indicated that attention was beginning to be organized around the skills and experiences needed to do interesting things with Logo as a programming environment, rather than discussing Logo as a vehicle for learning general cognitive skills. For example, three months into the school year, Jeff expressed his goals for the children in these terms:

> My underlying goal is for them to learn to put together programs of greater and greater complexity, to be able to achieve more and more with programs.

By November, Jeff reported that many children were having difficulty understanding important programming concepts: recursion was vague to most; variables were hard and seldom used; subprograms were not used effectively. The learning was child-centered: individuals were assigned to work twice a week and during these periods were free to explore the Logo language. Jeff worked with individuals, occasionally suggesting a technique or variation. There were no group lessons, and he did not push children to move on. He also provided written sheets and cards for the children, which served as mnemonic devices for certain techniques. Occasionally he gave them a complete program if he felt it was easier for them to use it as an unanalyzed tool. The children sometimes adapted these programs to suit their own purposes. Jeff recognized that the children often chose projects that were too ambitious, and he tried to help them to proceed more gradually.

In February, Jeff offered some programmed games to the children and suggested that they try to make some modifications. This strategy for learning did not work well because the children preferred playing the games to playing with the programs. In addition, those children who were sophisticated enough to make interesting modifications in

their programs were already involved in other self-development work. At this point, Jeff attempted some group lessons on special topics, such as random numbers, but after a few sessions did not find this format particularly successful for getting kids involved or helping them to develop new skills. Jeff also began teaching the use of xy coordinates, conditionals, tests, toplevel, and addressing memory locations to selected individuals. He encouraged children to get involved in certain kinds of goals, such as animation, guessing games, and word games. He reported that he had no class curriculum in computers: the children who were interested kept making new things; the children who lost interest did very little.

By the end of the year, Jeff described his goals for Logo entirely in terms of computers: "My goal is mastery--being able to manipulate and make sense of what to do with computers." Similarly, Dan expressed the value of Logo in a more limited fashion than he had at the outset of the experiment: "It's good for teaching about programming."

As a result of their increasing awareness of the complexity of the language, throughout the year both teachers struggled to understand what, cognitively, was being asked of the children in the course of learning Logo, and what was the value of devoting significant amounts of time to its practice.

> In teaching subjects I generally have a clear idea of what kind of skills I expect them to be learning, and in giving them specific work I have a good sense of what kids are able to do--analyzing and assessing their work. With the computer, it's still at early stage of development in my thinking. It's very difficult to understand what is going on with kids' understanding of computers, and this is true for kids who are very good with it and those who are not very good. I haven't yet built up a way of analyzing what happens when I give them a task, and what it means to succeed. I will think that the next step will be too hard and it won't be, or I'll think that it'll be simple and it turns out to be difficult. It can be very unnerving because no matter how good the kids' work is, there's a lack of understanding on my part of what really they're doing and where they ought to be going with it. [Dan]

Thus, Dan continued to revise his Logo program throughout the year. Midway, he added the element of "optional lessons" to his learning structure--giving occasional lessons on a special topic to a small group of children who expressed interest (e.g., how to do a variable). The technique was demonstrated, documentation sheets were

distributed, and the children who participated could then diffuse the information to other members of the class. Teaching was done individually or in small group, based on expressed interest. Large-group meetings were considered inappropriate because Dan believed that individual learning styles varied. At this point, like Jeff, Dan began to give children programs in Logo that they could use as tools in their own work:

> One thing that's really changed is my thinking about giving the kids programs. This came about with the read character program, which seemed like a very complicated program to just hand to the kids. I didn't know at what point they would understand what was happening until I showed the program to one kid who was not terribly concerned with understanding what the syntax meant. They were very capable of understanding how to type it out and which parts needed to go where, and therefore had their own understanding of how the program worked.

Computer sharing was introduced as a method of communicating information: the class began to meet as a group, and individual children presented and explained their work.

In the spring, Dan noticed a loss of interest on the part of many children, and began to work harder at scheduling time at the computer for all of them. He felt that sequenced presentation of material was not possible because the children were working with a variety of things. His support consisted of helping children to solve local problems, and giving them ideas for moving ahead with their projects. The focus of the small-group sessions switched from demonstration of decontexted techniques to demonstration of programs embodying specific techniques. As noted earlier, Dan had recognized that a major problem for the children was "putting the pieces together." He therefore decided to show them whole, working programs in order to give them some idea of how the techniques functioned in the achievement of goals. He also began to work with the children to help them connect a series of programs into one superprogram.

> A major part of what I've been showing them is how to connect a series of programs, how to think about joining specific things together to make a big program. If they can think about one thing at a time and break it down that way, it'll be easier for them to think about a whole program and put it together. We've worked this way on animation and how to put a story to it.

Dan also tried to build links between the programming work and the math curriculum:

> The Apple can use "setxy" commands to draw lines. You can vary the shape by varying the coordinates. I have been giving them experience with coordinates which is also part of the 3-4 grade curriculum, so I'd be doing it with them anyway. I want them to have this experience so that they can have it as a tool and as something that is part of the curriculum. We can do this as part of the math curriculum and also to advance their knowledge of the computer.

At this point in the year, Dan was unclear about how much the children understood, and therefore about how he should be helping them.

By the end of the first year, Dan recognized the necessity of and had accumulated enough experience to develop a theory of instruction. Some of the problems of the first year's work were revealed in his account:

> I was teaching programming skills piecemeal, and expected them to find a use for that skill. I hoped they'd come up with the picture. I realized near the end that I should show them the results of having the skill and then see what they do with it. After the demonstration I would say, "You have to do such and such."

Based on his experiences with two classes of children, Dan began to make his theory of instruction for Logo explicit. Both he and Jeff decided that there was a need to teach specific skills in a coherent sequence (e.g., what is a program, what is a variable, recursion, conditionals, tests); to context the skills in sample programs; to provide support for further developing their ideas; and to come back and talk about results. Encouragement to practice the skills was seen as important because

> It was rare for children to go beyond. They somehow stay pleased with things as they are, the passive approach. They want immediate results, so they pick small goals and are pleased with themselves.

Dan also decided that he wanted to include information about the history and function of the technology in the general culture as a part of the programming work, thereby offering a perspective for the children that would help them to see how this work fit into the world.

By the end of the year, Jeff reported that it was difficult to have group discussions about the computers because the uninterested children were hostile to the idea. Some didn't like the precision required for doing things on computers; others were interested in spending their time elsewhere. Throughout the year, Jeff had attempted to provide support to the children as needed in order to advance their work. He estimated that about 85% of the teaching was done individually or with two or three children. During much of this time, Jeff was committed to the idea that the children would learn Logo most successfully through developing their own projects. His efforts to engage the uninterested children included project suggestions, teaching skills by demonstrating interesting programs, and giving children programs to modify.

Unlike the younger children, computer sharing was not popular among the older group: "It just didn't connect with what they wanted to do." Because he found that disk housekeeping was a problem--no one could find their work on the disks--Jeff resolved to teach good saving techniques and documentations during the second year. He also felt that teachers needed a good deal of support in order to carry out the Logo agenda:

> The skills a teacher needs are monumental. The kids says, "I'd like to do this," and sometimes I have no idea how to do it. So I said, "Let's think of a simple thing, a part of it." I was afraid that was dishonest--I had that happen many times. We need to get expert help. It's not like helping the kids to write a story, where I'm never at a loss.

The First Year: Reflections on Cognitive Skills and What the Children Learned

During the first year, the teachers shifted their focus from Logo as a learning environment for general problem-solving skills, to Logo as a context for learning about programming and computers. One reason for this shift was that the teachers were accumulating evidence that caused them to reinterpret the value of Logo--evidence that can be divided into several categories:

1. There were large individual differences among children in interest and skill with Logo.

2. Many children had difficulty "putting the pieces together" into projects.

3. Many children reached "plateaus" of skill and didn't seem to be motivated to advance.

4. Many children had a relatively shallow grasp of the functioning of the language and were not flexible in applying what they learned in one context to a new problem.

1. Individual differences. Both teachers noticed and were concerned about the wide variation in the children's commitment to working with Logo. At the end of the year, both teachers estimated that about one quarter of the children were very involved in the work, one half were moderately involved, and one quarter were not at all interested. Although not singularly true, the teachers were concerned that many of the unenthusiastic individuals were girls (see, for example, Hawkins, 1984). Given their expectations about Logo's broad appeal for children, the teachers were surprised at the effort required to motivate children. They had great difficulty in getting the uninterested children involved in the work. In this effort, the organizing idea was to help children find a project or goal for themselves as a means of getting into the system; the learning of skills was subordinated to the articulation of goals.

> Another problem is that not every kid has a project that they're working on--that was my goal at the beginning. I haven't insisted because of practical pressures. Also because some of the kids need a lot of play, a lot of directed play...I was heading for everyone having a project real soon, but I've changed that. What this is is my learning about the process. [Jeff, in November]
>
> Trying to get kids who are not interested in projects on their own to get interested--that has been my struggle all along. I've been trying to get them to find activities that they're interested in. [Jeff, in February]
>
> My biggest task is to get kids interested who aren't doing very much. It's similar to writing in that I focus on helping kids to develop skills. [Jeff, in June]

By the end of the first school year, both teachers were beginning to voice concerns about the possibility that, in contrast to the idea that Logo was, per se, an exciting and beneficial environment for all children, certain prior skills or interests might be prerequisites for enthusiastic engagement with Logo. For example:

> A lot of abstract thinking is required. I wonder about limitations on people's ability to do programming. Maybe

> some people can do it and some can't [do it] as well because of the highly symbolic abstract nature of it. It may be a misconception that people need to learn how to program computers to survive in the world. It may be a very limited field, more inaccessible than I thought--maybe it's not like reading that anyone can do....I have no impression that it changed the thinking of kids, but I don't feel that what I did with computers was particularly successful, so maybe it wasn't a fair test. And it could be that the kids that were good were already thinking that way. [Jeff, in June]

> Kids who do logical thinking develop programming skills. I now think that the reverse is unlikely. [Dan, in June]

2. Making programs from pieces. From the beginning of year 1, Dan was particularly concerned with his observation that, although many children learned individual commands or techniques, they had difficulty putting them together into coherent programs. In other words, they had difficulty using the symbol system functionally.

> Even though they have all these pieces of things that they know they can do, it is very hard for most of them to put all these pieces together--it only happened in two or three cases. [Dan, in October]

> I began to demonstrate actual programs to them rather than just techniques. The kids weren't putting the pieces of the techniques together to make wholes. Some kids, yes, but for the most part, kids were having a hard time seeing all the pieces and taking the big step. And even when they do, they'll get satisfied and go no farther. [Dan, in February]

Jeff also reported that putting commands together to construct programs became increasingly difficult for many children as the year progressed and the programs became more complex. Translating an idea into the code required to execute it was a difficult task for both children and teachers. The children often changed their goals to accommodate the program, rather than trying to rewrite the code in order to achieve their original goals.

3. Plateaus of skill. Both teachers reported what they found at first to be a strange phenomenon: Many children reached a "plateau" of skill, sometimes quite early in their learning, and were content to continue constructing programs at that level. The teachers then realized that this could be said of many areas of the curriculum, but

that it was especially true of the programming work with its individual-project focus. This was particularly demanding of the children because it required them to be self-motivated in learning about and adapting new tools to their own purposes. The teachers directed their efforts at encouraging the children to do variations on their work, and offering them new concepts that they might incorporate into their work and thereby gain new skills.

4. Depth of knowledge. The teachers also found that most children did not have a very deep understanding of many of the commands or the structure of programming, and thus were not flexible in applying particular commands or concepts to new problems. As discussed above, the teachers began to realize that it was increasingly unclear to them just what "full" understanding of the language entailed--how deep did the children's understanding of the functioning of the system have to be? As they struggled with this problem, both teachers expressed as their overall goal that children be able to use the concepts flexibly. The core of their implied definition of expertise was that the children be able to accomplish the things they themselves wanted to do in programming the computer. However, by the end of the year, most children were well below this level of ability.

> Some kids know commands as nonflexible units; others can use the same command as a tool. I see differences between kids in using a command as a sophisticated tool and using it without clear understanding. [Jeff, in February]
>
> There is a gap between what kids want to do and the skills they need to do it. Some programs have to be very complex to do simple things. [Jeff, in June]
>
> I would like them to understand the instructions, that if the kids had an idea they would be able to do it. [Dan, in February]

The First Year: Reflections on Self-Initiated Learning

As noted above, the second idea which guided the incorporation of Logo into the classrooms was that it would support self-guided, expressive learning. Logo was an environment where children could learn complex, logical skills through a medium that allowed child-centered expression, as opposed to teacher-directed learning of component skills. Although they made significant modifications, the teachers remained committed to the organizing framework of child-selected Logo projects throughout the two-year experiment. The

following account summarizes the issues they struggled with as they reinterpreted the value of Logo as a tool for self-initiated learning.

During the pilot period, the teachers had begun to realize that the children were having difficulty developing their own goals for Logo--a considerable problem for a learning experience firmly rooted in self-motivated expression. But they were firmly committed to the strategy that the children be given access to Logo and then make their own decisions about what they wanted to do with it.

> I see the computer as another area of the curriculum, an area that it's very much up to them what they want to do. I would never want to give a kid something to do on the computer that the kid would say, "No, I don't want to do that." I'd rather it remain optional until they have adequate command of the language....I would like the kids to be motivated to learn, so they can decide if the computer is something they want to express themselves with. [Dan]

Throughout the first year of the experiment, the teachers struggled with the problems involved in maintaining the self-initiated framework for learning, while becoming increasingly aware of the complexity of the learning skills in this domain. Many children had to struggle with both the computer skills and finding their own goals for the work. This theme remained a constant and unresolved tension throughout the experiment.

> I didn't give them a basic introduction. Basically I wanted to get kids thinking about a project. I'd leave that up to them, but I realize from last year that they really need a lot of structure for their thinking about what to do on the computer. Their imaginations seem to stop at a certain point. It's sort of shocking for me to see that. [Dan, in October]

But the self-initiated orientation was valued by the teachers, who saw it as especially powerful for those children who became engrossed in the programming work.

> With their programs I have rarely said where I want them to go. Part of the reason is that kids have their own ideas of where they want them to go. It generally happens that kids say, "I'm done." Kids don't say that with computers. If they do a program, they want to play with it and improve it. It's a new material and a new type of work. It's more self-directed and kids can work absolutely alone. They might ask more difficult questions though, and that's

> something that doesn't happen in other parts of the curriculum. [Dan, in February]

By June, Dan was expressing more need for structure and teacher input to children's programming work:

> The best results were in the last two months when I insisted on a project, and my having input into it. More people were interested and they asked more questions. It's not a tool you can just hand over to kids and have them express themselves, as I'd expected.

Thus, while maintaining a commitment to the self-initiated framework, Dan was beginning to interpret Logo less as a context for solely self-guided expression, and more as requiring some structured guidance from the teacher.

Jeff also reported that children were having difficulty in formulating goals for themselves, and that some simply abandoned the work. He found that he had to carefully think through what was appropriate for the children to attempt, and then develop ways to support the work they chose to do.

> I've been having them work at the computer exploring the possibilities of computers, and not giving them specific tasks, encouraging projects of various sorts. If I see them working on something, I might ask them if they'd like to try such and such, and I would teach them a technique. I haven't pushed the kids to move on....I am realizing what is feasible and what is not. In the beginning, kids took on things that were too ambitious. Recently, I've been more careful about getting them to go little by little. I am more conscious of what constitutes a feasible project.

By the end of the year, Jeff felt that the fully self-initiated learning framework worked well for only a few children:

> If they developed a good project, the work was enjoyable and intellectually challenging. But not many kids did that sort of thing.

Significantly, Jeff added, "I didn't teach it in the right way." Jeff's interpretation of his role in the enterprise had undergone significant revision--from a tool which "taught" itself, to a topic which required careful teaching.

By the end of the first year, both teachers were uncomfortable with the child-centered, self-discovery pedagogy embodied in the Logo rhetoric. They felt that the radical version of this approach--child-initiated goals with support from the teacher when appropriate--had worked well for only a small number of children, which they attributed to two factors: First, the children had difficulty developing goals in this new medium. Unlike other domains, they brought little world-knowledge of what could be done with Logo. "Social studies was much more contexted than Logo programming--in social studies children have ideas already about what it's possible to do." Second, as noted above, the teachers were struggling with the problem of what it was necessary for the children to know, and of finding the best ways to support this in the group-learning environment of the classroom. Both teachers were disappointed with the results of the experiment during the first year, and saw their own lack of insight and skill as being at least partly responsible.

Summary of Year 1: What is Logo?

At the end of the first year, a critical question was being asked by both teachers and children in the two classrooms: Is Logo a legitimate part of the work of the classroom? For the teachers, this meant two things: First, is their experience with Logo teaching the children important concepts and skills? Initially, the teachers had believed that the children might learn general problem-solving skills. This belief gave way to doubt about the generality of such skills, and a new question emerged: What is the importance of computer programming itself as a topic?

The second interpretation of the legitimacy of Logo concerned the way Logo was taught. How does it relate to other areas of the curriculum? Is it a "real" subject, and what instructional requirements and techniques are appropriate?

> Even in parts of the curriculum they're very excited about, I have to direct them. I have to give them a sense of what's next. With a lot of kids they can develop a sense of what's next with the computer, but for the kids who aren't good with the computer, the decision as to whether to learn it is very personal, and until I make the decision to make it part of the curriculum, then it will remain a private area. It's now a self-directed experience. [Dan, in February]
>
> Next year it'll be a subject kids are responsible for. I'll teach particular skills, talk about results, concept-oriented. [Dan, in June]

> It's different. There haven't been failures because I haven't imposed goals on their work. It's completely child-directed. They're doing it, but I wouldn't force them to do it. My goals are very flexible, very different for each child, much more tailored to what they like. My goals for the academic work have been much more uniform. There is a sense of direction in academic work, uniform direction that there isn't in computers. [Jeff, in November]
>
> One thing I'm struggling with is, what is my role vis-a-vis the computers outside the individual format? The other thing is how do the techniques which I use in other teaching apply, are they relevant? [Jeff, in February]

The work of the children reflected the ambiguity of the status of Logo in the classroom. For them, the question of its legitimacy concerned the lack of clarity about what the activity was, and their responsibility for working at it. The radically self-initiated style of the work was uncomfortable for many children because it was so different from the more structured tasks of the classroom that they were used to. There was no formal presentation of material, no group lessons, no task requirements, and no evaluations. Because the children were not sure of the legitimacy of the Logo work in the school environment, they didn't how much effort they should give to learning it.

> There were kids who were losing interest. They were receiving a double message. I wanted computers to be part of their work, yet it was optional. Kids would reach a problem in their work, and they would be less inclined to push through it in the way they might with other work because I didn't make them. [Dan, in February]
>
> Some kids felt, "Isn't it funny we can just fool around with this?" [Dan, in June]

Jeff also reported that some "kids didn't feel that it was a legitimate part of their work. Others saw it as an intrusion."

Thus, over the course of one school year, both teachers began to assimilate Logo into the complex pattern of learning that was established in their classrooms. Logo-out-of-the-box was problematic because teachers found it to be too complex for children to master without systematic support. The radical self-discovery pedagogy of Logo gave it an ambiguous status vis-a-vis the more structured learning tasks of the classroom because it was unclear where the

responsibility for getting the work done rested and, indeed, what the work itself was.

By the beginning of the second year, Logo learning began to take on some characteristics of learning and teaching in other curriculum areas. However, the teachers recognized the overall nature of the interpretive problem they were struggling with:

> I have a clear idea of culture such as it is, past culture, of what it means to be an educated person in terms of all different subject areas. In terms of the computer, it's not part of our culture, it's something new so I don't have clear ideas of what kids should or should not know. In a sense, I can be much more conceptual about computers than I can about other kinds of work because society as a whole doesn't know yet what it wants to do with computers. [Jeff]

Section II. The Second Year: The Development of an Instructional Strategy

During the second year of the experiment, the teachers began to provide a particular type of structure to Logo learning based on their experiences with effective education in other subject areas. Unlike the first year, when the teachers were committed to adopting Logo in the manner advocated by its developers, the second year saw Logo being adapted to the shape of the existing classroom learning context. Logo was also offering new perspectives to the teachers about the teaching of cognitive skills. While they viewed programming as a "subject" requiring precise, analytic skills, the classroom work continued to be seen as more like social studies and writing and less like math and grammar; the core of Logo learning remained organized around self-initiated programming projects. Because of their belief that this sort of analytic skill required a structured, sequenced curriculum, both teachers developed a parallel "track" where concepts were "formally" taught and children were held accountable for learning certain concepts and doing assigned tasks.

Over the course of year 1, Dan's thinking about learning Logo underwent a major shift in focus from a belief that the structure of Logo naturally supported children's discovery of programming concepts with little instructional intervention necessary ("In general we use the organization that's natural to the computer"), to an effort to develop a means for systematically supporting the learning of important programming concepts.

The Second Year: Revisions for Classroom Work

By the second year, Logo was seen as a tool for learning about computers, with the emphasis on programming. There was no longer any talk about the acquisition of general problem-solving skills. Although there was still some ambivalence about its ultimate value for all children, by the end of the experiment both teachers felt that learning programming in Logo was valuable for many children:

> Programming can be a good idea, to give kids control over the computer, how the computer is working, invest their own ideas in what is happening. There were few kids who I thought it was totally worthless for. [Dan]

Jeff also felt that Logo was valuable, even if no broader goals were achieved through its practice:

> My thinking now is that what the thinking kids have to do to make up programs is a worthwhile, useful exercise. I don't know about its application or impact on other thinking, but even if there were none it wouldn't bother me. So I'm giving them skills, making sure they understand them.

Jeff decided to work out a sequence of skills to present to the children. His concern was that children missed concepts, or misunderstood some things and thus could not proceed. Therefore, he began the year by assigning the children lessons and tasks for practicing these skills. At the same time, the children were required to develop their own projects.

> This year's structure is designed to teach skills. Kids who are facile with the computer can do what's assigned in five or ten minutes, and then do their own projects.

The assignments were designed to ask children to use concepts in contexts similar to, although not identical with, the ones they were taught. Jeff felt that these assignments would be diagnostic tools to help him understand what was going on with knowledge development. Jeff continued to find it difficult to help kids develop their own project ideas, to give them "germs" from which larger projects could grow.

Requirements for documentation were laid out, so that the children and Jeff could easily find their programs. Children were required to keep a card file of their work.

At the beginning of the second year, the teachers were no longer defining Logo primarily as a medium for self-expression, but as a curricular topic with particular concepts and skills. Both teachers devised plans for the second year's work:

> My plan is to teach programming skill in sequential order. Last year I taught things when kids needed it or were bored; this year I'm teaching in a particular order, it's more coherent. It's arranged so that the computer is work. [Dan]

> This year there will be a weekly computer period. I will teach skills in a directed way, with equivalent importance to other subjects. There will be less time just free with computers in classrooms. I'll work out a sequence of developing skills. I'll teach record-keeping techniques. [Jeff]

Thus, in the second year, like Jeff, Dan decided on a sequence in which he presented Logo concepts, and held weekly lessons for the entire class. The lessons were accompanied by worksheets that the children could use for reference. The children were required to develop their own projects, but Dan also gave them programs that he asked them to look at and modify. In effect, there were parallel learning activities going on with Logo which sometimes intersected. Children learned skills through the formal lessons, and worked on their own projects during assigned work periods. Sometimes the new information was incorporated into the ongoing, self-initiated work, and sometimes it was not. Dan was hoping to develop a conceptual framework for the children and an organized repertoire of skills that they could draw on as needed. Dan continued the practice of embedding techniques and concepts in working programs, as well as the group sharing where children demonstrated and explained their work to each other. He recognized the need to draw children's attention to the details of a program in order to develop functional capacity with a technique:

> I think that by giving them more of an opportunity to think about what's happening and see how it works, we can make those difficult steps. I've been focusing a lot on looking at the program and really reading what I'm putting in front of them--to think about what it's going to do before it runs. Thinking about how I demonstrated things in the past, I'm sure I typed things on the screen and told them what was going to happen and then did it--not making sure they're reading the words, not making sure they know the signifi-

cance of each part, and not giving them the opportunity to practice before they've gotten it themselves.

Dan felt that it was important for the children to be able to share their work and knowledge publicly, that such opportunities added to the importance and coherence of the work for the children:

> So I found that I had started giving them reinforcement in the way that I do with other subjects all the time but sort of never really did with the computer. The thing that gets the most response from the kids if you're teaching is still to give everybody the opportunity to give an answer. So that if you have a wide open question where anybody can contribute anything to it, kids will always respond--that's how we try to teach a lot of things. Allowing kids to read through a program over and over again--what's this line going to do. Five or six kids will raise their hands because they want to be able to say what it will do, and they feel good that they can say it and nobody objects.
>
> They're very interested in sharing something. The importance is not so much the product but in saying I worked on this and I want to show this to you. Same thing happens with computers as happens with their stories. They get very involved in it. They pay very close attention to each other's work. They don't say "no" to each other's work because there's too much of a personal connection. They have a better understanding than adults do sometimes. They can empathize with the kid.

The learning experience in the classrooms during this year was designed to address some of the problems that the teachers had observed during the first year's work. In November, for example, Dan discussed his intention of helping the children to deepen their knowledge of Logo in order to enable them to use it flexibly:

> They can't understand why what's happening is happening. So they are working out their own problems, their own programs, but they're not getting much out of it. I want to make what they know more coherent to them. Like last year, Kathy made a program with eight variables but it wasn't clear to her what she was doing. Kids don't ask why things do what they're doing. In that program she would just throw anything in. There was nothing systematic about figuring out why those numbers worked.

Dan reported that this lack of pursuit of understanding was not limited to learning with the computer:

> Function is so important, so much more important to them than understanding. If they're reading a story, and they come across a word they don't know, it doesn't matter to them if they can still understand the story. But if it's a focal point in the story, hands shoot up right away. Kids won't ask why things are the way they are in math. If they're working on the abacus and I say that 8 times 7 equals 56, no one asks why.

Thus, Dan decided that programming was analogous to other subject areas, where teacher support and direction were essential to the development of depth of knowledge.

Jeff also cited the development of deeper understanding as a goal of his work with Logo:

> A major concern from last year was that kids didn't understand things and that inhibited their performance on the computers. This year I'm going to be more thorough in the instruction I provide so I help kids who would otherwise drop out--the kids who missed concepts and couldn't proceed. I want to make sure the kids get the skills to build upon.

An additional problem that both teachers noted during the first year was the difficulty children had in putting the pieces/commands of Logo together into programs.

> With the computer, it's an arbitrary logic. It's very difficult for them to put the pieces together on their own. It's hard not to function as a mediator.

Thus, they began to present programming techniques in the context of working programs to demonstrate the function of a piece within a whole.

In order to deliver sequenced information to children about critical programming concepts, the teachers set up situations which encouraged children to think through in detail what was happening as programs were constructed and run, and they helped children to develop project ideas. The notion of longterm projects selected by individual children continued to be a dominant framework.

By the middle of the second year, Dan was talking about the organization of learning in terms of the kind of structure that was needed to help the children acquire difficult concepts. He was no longer reliant on the "natural organization" of the material, but recognized that he as a teacher had to organize the presentation to help kids see clearly and avoid misconceptions:

> Whereas last year I would probably have introduced it [variables] as a technique, the past few months I've been trying to think about how they would understand it best. Not just from how the computer works, but how they would understand it best....It's building up each step logically, what you need to have to do before you do the next thing.

The teaching of Logo was beginning to be interpreted as analogous to other subject areas where a teacher has to structure the material in accordance with his or her theories of children's learning:

> It's forced me to sit down and think, well, I have this sequence of skills, and what comes first and when...it makes me think of how to flesh out each concept.

During this year, Jeff's formal lesson structure broke down because of the limited time he had available for teaching, and because of the resistance of the children to these lesson formats. Jeff also began to feel that there was little relationship between the lessons and what the kids were doing on their own.

> I realized that the lessons were too rigid, meaningless to some because they were too difficult. For others they didn't fit with what they wanted to to.
>
> When I gave lessons or taught techniques, many kids didn't care if they understood or not. Some wanted to do sophisticated things but I didn't have the skills, so I taught them pieces....The sequence of concepts didn't work. It was too fast for some and too slow for others. It went from drawing to relative direction, commands, absolute direction, variables, recursion, changing variables, conditionals, print. I realized the lessons weren't faring well when there were no interesting projects.

Jeff reported that, for the second half of the year, he gave no worksheets and assignments. Techniques were demonstrated--"short things to learn"--and individual projects were again the major emphasis. Jeff felt that it was a mistake to try to get all the children

"moving at the same speed": "Programming is like math. You could not teach a group of 26 children all the same math."

At the end of the second year Jeff was again disappointed in the year's work. He felt that few interesting projects had resulted, and was concerned at the number of children who had "dropped out" of the programming work. Like Dan, he was not sure how to bring together the disparate elements of learning Logo in the constraints of the classroom: Logo should be based on functional learning through an individual project focus, yet the children needed a framework to master difficult concepts and skills in order to do their independent work. He found the task of helping children to identify appropriate projects to be difficult and frustrating. Because the children learned in different ways and at different speeds, whole group lessons were problematic. In addition, Jeff continued to feel that the legitimacy of the work was still an issue among the children.

> There were changes from the previous year. I tried to make it clear that the computer was a subject. I set aside periods for it. But even this year, few children thought of it as a subject. The older kids knew about tests for schools, and they talked a lot about that. They're not tested in Logo like they are in other things. If I did it again, I would do something like evaluation of skills. I would do more group teaching.
>
> There are some kids who still have trouble, and obviously I have to continue to think about how to connect them. There's an enormous range, from kids who are very good to kids who can barely make a program. Some kids pick up an idea and do all kinds of things with it. They explore. Others, it's as if it truly is an alien medium to them. [Jeff, in May]

By the end of the second year, Dan felt "like the pieces are fitting together." He began to observe that the children were using skills they had learned earlier in the year in both individual and group projects. For example, one group of children was creating a solar system program that used a lot of different techniques: "They finally found a reason to use some of the things we'd been talking about."

While he felt that there were several different levels of ability represented in the classroom, Dan also believed that proficiency and interest was more widespread than it had been the previous year. He characterized the year as one in which he had been doing a curriculum and group lessons, and didn't have much time left over to individualize things. The promise of Logo had been just that--a learning

environment where the pace and sequence would accommodate very different individual interests and needs. When it became clear to Dan that the language was too complex for the children to learn without considerable structured support, he focused in the second year on a way to offer that structure to the class as a whole. The issue of how to provide an overall framework and yet deal with individual levels of understanding was an unresolved one for Dan in light of time constraints in the classrooms.

> I feel that I have to have more time to figure out ways of involving them all in solving a problem. There was always a clear difference among children but I wanted to ignore it for the sake of time. It's very hard to get away with not differentiating--I don't like to do it. I've never used grouping where they're grouped according to skills. The first time I've ever done it is with math, so maybe I should do computer when they're already grouped that way. When I get more abstract now, fewer kids can follow. I want to make time for the good kids--some kids are so good at things, I'd like more time to work with them individually just to see what's going on. For less good kids I need to figure out ways to help them--to see what's going on and figure out ways to get them to participate, not necessarily in the group, but in doing something productive for them--that's still the hardest. To work with anyone on the computer takes an enormous amount of time. That's why I'm talking about splitting them up into groups. I see there's a real need to focus independently on the kids who really know the skills, and then to meet in some way the needs of the kids who don't know the skills. I've been working on a sophisticated level, sort of pushing them. That means the kids who can't get pushed are lost.

The Second Year: Reflections

Throughout the second year, the teachers reflected on their role in relation to Logo as they tried to assimilate it into their classrooms in a structured way. Their original perceptions of what Logo, as a fully self-initiated learning experience, could offer to children, and their subsequent awareness of the need to make it a structured, legitimate, and accountable part of the classroom work created a tension about their understanding of their own roles that continued to be an issue at the end of the second year. During this year, Logo was defined by the teachers as a new curriculum area, thus giving it legitimacy in relation to other classroom work.

Dan felt that his commitment to sequence and structure in the learning of Logo helped him to better understand what was going on with the children. He reported that he was forced to think through the requirements of learning the language. Dan also felt that his commitment to providing a coherent learning structure allowed the children to feel more comfortable; Logo was defined in terms that legitimized it in children's eyes as part of their appropriate work in the classroom setting.

> It's different from last year, primarily in my perspective. I feel like we're going somewhere. I don't know if the kids feel the way I do. I feel that they're going to end up with something. [in November]

> This year it seems less of a problem to help kids figure out what to do on the computer--in fact they really seem to have ideas. I think it has a lot to do with my attitude toward teaching it, that it is a subject I expect them to be responsible for, a lot to do with the coherency with which things are taught. It's forced me to sit down and think--I have this sequence of skills, and what should come first, how to flesh out each concept. It's building each step logically: what do you need to have before doing the next step? [in January]

Dan reported that throughout the year, most of the children were more engaged in understanding the details of program construction than they had been during the first year:

> The kids demonstrate what they're working on and they explain it to other kids....Kids will ask, "Let me see the program." To me it's a sign that they're thinking about how it's happening, that they would ask. It didn't really happen last year that kids would ask what it looks like.

At the end of the year Dan summarized his strategy and agenda for the children: He had decided to lay out the programming concepts and techniques in a sequence, and present them to the children in meaningful contexts. He discovered that it was best to introduce each new concept as part of a program that interested children--a problem situation. This gave the children an idea of how it worked in context, and what it was good for. He felt that it was important to combine two things--presenting concepts in a clear way, and doing problem-solving work with the children. Self-discovery-based learning was problematic because the ideas inherent in Logo needed to be clearly and carefully pointed out.

> This year I had an agenda, and let them take part in it as they wanted to. Last year I was more involved in their individual work--seeing how I could help them rather than giving a general framework. It was more hectic last year, giving pieces of information that did not fit together.

Overall, Dan was pleased with the development of this teaching strategy and the engagement and accomplishment of the children during the second year. However, he still pondered the paradox inherent in his role in this structured yet self-discovery-based learning situation:

> Sometimes the kids have the right idea of what ought to work on the computer but the syntax is so exact--the right idea but the wrong set of words, and there are no books that are useful to them. It's a very funny situation because it really makes a myth out of what Papert is saying. Even though kids feel very much in control, very much involved, it's made my role as a teacher much more traditional in a funny sort of way. I've tried in the lessons to give some idea of the structure of the language, but still I'm the dispenser of the information. I still have to say, "This is how it works." It's different in other areas--I can ask them a question about how people coming over from Europe in 1600 would manage to live, and from the knowledge they have and from speculation they could come up with some very good ideas and they can have a discussion. We can't really do that with the computer now. That's actually something I'm trying to do now--thinking of more open questions that have to do with computer logic but not necessarily with the exact answer. [in June]

Jeff was less satisfied than Dan with the Logo experiment during the second year. This was perhaps partially due to the fact that Jeff was on half-time paternity leave during the year and had to accomplish a great deal during his limited time with the children, and partially to the fact that he found it difficult to engage many children. Jeff found that the structured sequence he had planned did not work well--many children were resistant to organized lessons about Logo. Throughout the year, a dominant theme in relation to his teaching role was the large individual differences he saw in interest and ability, and the conflict between letting children follow their own interests and holding them accountable for learning the programming skills.

> This year I tried to make clear that the computer was a subject, and set aside periods for it. But even this year, few kids thought about it as a subject like others. They

weren't tested in Logo, and they know they have tests in other subjects. Many kids didn't like the lessons, but when there were few organized things, kids dropped out. [in June]

Jeff found that he had difficulty providing both organized lessons and the project ideas which grew out of these lessons as germs for long-term programming commitments. Jeff continued to grapple with the need for some structured learning, and the fact that individual children assimilated and made use of information at different rates. Like Dan, he remained committed to the organizing framework of individual programming projects where children pursued their own goals as a means of learning. But he had difficulty with organizing the information-giving so that children could make use of it in their own way and at their own pace.

Another part of my approach is thinking that if I just let kids fiddle and play with it, they'll run out of things because they don't know what to do. So I wanted to give them techniques, project ideas. And they were very conscientious about it--they always did what I gave them. There were kids who got techniques out of it, but there were kids who didn't understand, so my going on with what I thought was this curriculum was a problem. In math, I have a curriculum, and the fact that they don't perfectly understand or can't manipulate all the concepts doesn't mean that I don't go on because they do get some of it. The difference with computers is that having experienced it isn't enough to apply it to your own work, just as an artist who had some exercises in perspective couldn't use it unless he or she played with it a lot. You need much more experience than just a lesson with it. [in May]

So what I decided about three months ago is that I couldn't teach a whole group--that just didn't make any sense at all. I had decided in the beginning of the year that I could do that. I wanted to make it into a piece of the curriculum, into something that the kids felt was part of their work. Doing it in this very "projecty" way, kids following their own interest, I was having a lot of trouble making them feel that this was work they had to do. Most things aren't run that way--most classroom activities are much more assigned. So I felt I had to structure their work more and give it direction. What happened was that in giving them direction, I was trying to get them all moving at the same speed and that just wasn't working. It's not like social studies--I would never divide the class into more and less mature

> groups because the thinking about people, even if it takes place at different levels of maturity, has important commonalities. There's a core of understanding about what a person is that abstract symbolic information lacks. [in June]

Thus, at the end of the second year, Jeff also confronted a paradox: his understanding of the necessary pedagogical structure for learning an abstract symbol system (leading to considerations of sequence, accountability, and ability grouping) was in conflict with a commitment to the importance of self-initiated, discovery-based learning. He struggled with a problem he saw embodied in the experience with Logo, namely, that perhaps this form of cognitive achievement was incompatible with a self-expressive learning mode.

> It's very difficult to tell what is going on with kids' understanding of computers, and this is true for kids who are good and those who are not. I haven't yet built up a way of analyzing what happens when I give them a task and what it means to succeed....[Consequently] I don't intervene much in their work.

Conclusion

The two teachers who participated in the Logo experiment revised their understanding of Logo as a learning environment as they worked with the students in their classrooms. Rather than viewing Logo as discovery-based learning, they saw it as a complex symbol system requiring structured support. Logo allowed new possibilities for students and teachers, and was itself shaped by the people who made use of it.