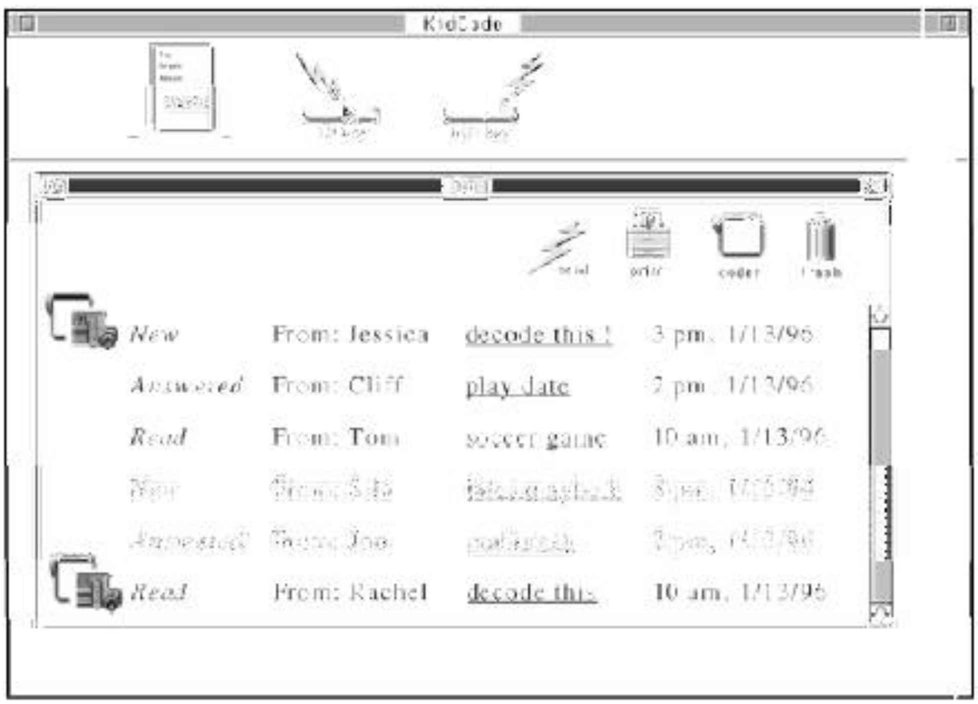
KidCode[1]: Using Email to Structure Interactions for Elementary Mathematics Instruction

Michelle Baker
Intellinet Inc.
325 Riverside Drive
New York, NY 10025
michelle.baker@acm.org

Judith Levy Cohen
Mathematics Department
Collegiate School
370 West End Avenue
New York, NY 10023
JudiCMed@aol.com

Babette Moeller
The Center for Children and Technology
96 Morton Street
New York, NY 10016
bmoeller@edc.org



This material is based upon work supported by the National Science Foundation under Award No. DMI-9561725. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

[1] *KidCode* is a trademark of Intellinet Inc.

Abstract

KidCode is email-based software designed to supplement the National Council of Teachers of Mathematics (NCTM) elementary curriculum standards by addressing the need to develop conceptual links between concrete mathematical activities and mathematics as a language. It provides children with the opportunity to explore mathematics as one of many symbol systems that have been built for the purposes of communication. We designed a sequence of four two-person games centered on the theme of codes and secret messages and conducted formative evaluation of the games.

By pairing tools for the creation of codes and authoring of coded messages with a full featured email messaging system, *KidCode* enables children to gain experience with many kinds of symbolic representations. Message authoring can consist of text, graphics, and even animations. Similarly, with *KidCode* children can advance from an understanding of symbolic representation of object/noun type entities for text and simple pictures to coded representations of spatial relationships and, finally, to representation of operator actions. Our work provides an example of how mathematics instruction can be based on the use of email software to structure cooperative interactions among elementary aged students.

For formative studies, we crafted materials of paper and poster board with varied examples at different levels of difficulty that could be used for multiple rounds of play. These materials were used throughout the research period in evaluations with three adults and twenty children in the first through fourth grades (ages 5-10 yrs.) from a wide range of socioeconomic backgrounds. As the research progressed, the materials were revised or expanded in order to address issues that arose during the evaluation sessions. Some of the games were modified considerably after observing children play the games. Our research suggests that, taken together, the games improve the children's facility with symbolic processing.

In this paper we describe one of the KidCode games - Rebus - in detail. We explain how formative evaluation based on paper materials was used to test design ideas for the KidCode software and develop a sequence of games to scaffold children's understanding of symbol systems and that would work effectively in an electronic mail context. The figure on the cover page shows the email inbox from a screen dump of a Macromedia *Director*[1] prototype of the software. The bus icon indicates a Rebus game message.

Background and Rationale

The ideas for *KidCode* arose out of studies of educational research that indicate that children's difficulty with mathematics in school arises from the failure to develop conceptual links between mathematics as a symbol system and mathematical concepts that can be used to describe the physical world. (e.g. Ginsburg, 1989, Goldman et. al., 1997) Whereas the development of mathematical thinking in young children and skill in basic counting and arithmetic can be quite advanced when posed in real world settings, their application in school math has seemed limited and often, resistant to further development.

Research on children's mathematics thinking has shown that understanding of basic mathematical concepts generally developed well before children's facility with symbolic representation. Preschool children, who cannot yet read are usually quite competent in using mental math to solve simple arithmetic problems requiring addition and subtraction. However, young elementary aged children have generally not developed an ability to use symbols. Few first graders are able to read upon entering school and their experience with symbolic processing is very limited. Nevertheless, until recently they have been required to use to use symbols to represent relatively abstract mathematical ideas.

The result of emphasizing symbolic math before the children develop competence with symbolic processing has been negative. This is where children begin to view school math as separate and unrelated to anything in their experience. Failure to develop children's facility with symbolic representations has the result that many children in early elementary grades develop misconceptions about school math as a nonsensical system of symbol manipulation. In studies of children who perform poorly in mathematics at school, it was found that the same children could solve problems with ease if they were presented in context and without recourse to paper and pencil. (Ginsburg 1984, 1997) Related research has shown that children's performance on mathematical problem solving tasks actually declined after a few years of schooling. Children upon entering school

[1] *Director* is a registered trademark of Macromedia Inc.

showed good insight relating arithmetic to real situations but by the ages of 9 or 10 were trying to solve word problems by resorting to superficial strategies such as "guess the operation" or "find the numbers and add". (e.g. Carpenter et. al. 1993)

Summary Description of the Research

Our main objective was to demonstrate the feasibility of computer software that allows young children to manipulate symbolic representations and communicate these representations to their peers. An important part of the work was to test and refine ideas for games that were intended to be self-teaching in the sense that corrective mechanisms including the incentive for evaluating one's own performance and the acquisition of relevant information would be built into the structure of the games. In addition, we wanted to develop a series of games that would cover the intended age range of first through fourth grades and build upon one another to help scaffold the children's developing knowledge. We aimed to develop a series of games that would not require intervention from a teacher, that would lead to more sophisticated skills with and understanding of symbolic representation, and be interesting or fun enough that children would choose to play the games on their own.

Overall, we have found that the theme of coding and secret messages is an excellent venue for learning about symbolic representation. The context of interactive communication is very appealing to the children we have worked with. They liked the idea that their communications were secret and were clearly amused as they anticipated the decoding of their messages. They found many opportunities for humor as partners misinterpreted some of the codings and we had many lively sessions in which coder, decoder and evaluators shared in frequent laughter.

An important advantage of having communication as an explicit theme is that misunderstandings naturally become an important source of feedback to the participants. Misunderstandings are less likely to be construed as failure but rather as useful information that can form the basis for message revisions. This is particularly important in mathematics instruction where even very young children have acquired preconceptions that there is always a single correct answer and failure to produce that answer is a sign of one's inadequacy. Finally, our research suggests that, taken together, the games help to improve the children's facility with symbolic processing.

In this paper we describe our overall findings, formative evaluation procedures, and give a detailed description of one of the games - Rebus. Our focus is on how the formative evaluation was used to develop the *KidCode* games and inform the software design.

Related Work

Software for cooperative learning is usually conceived as a tool for collaborative problem solving. In this approach to collaborative learning, software design issues include how to share input devices (e.g. Bricker et. al. 1995,), methods for sharing and controlling views of a computer display (e.g. Gutwin et. al. 1995), and means for communication among problem solving participants (e.g. Edelson et.al.1995, Guzdial et. al. 1995). *KidCode*, represents a different approach to cooperative learning in which interaction is not structured as cooperative problem solving but rather as communication per se. In cooperative problem solving, communication is essential but auxiliary to the participant's objective which is to solve a given problem. With *KidCode*, communication is the problem that participants seek to solve. In cooperative problem solving, learning occurs as children argue and demonstrate their ideas to one another. With *KidCode*, learning occurs as children revise and refine their understanding as they seek to repair misunderstood communications.

Because the focus is quite different, *KidCode* faces a different set of software design issues. For example, problems with sharing views and input devices do not arise in an email environment such as *KidCode*. Instead, some of the most important software design issues related to cooperative learning for young children are (1) how to provide sufficiently tangible interactivity so as to preserve a compelling sense of engagement with one's partner, (2) how to provide the coder with adequate cues to analyze and correct misinterpretations, and (3) avoidance of "technological artifacts" that could interfere with intended communications, e.g. through poor quality sound or graphics.

KidCode also differs significantly from other elementary mathematics curricular materials, whether software, paper, or manipulatives. Most efforts to address *NCTM Standard 2: Mathematics as Communication* have involved linguistic discussions of mathematical activities. (e.g. Scott et. al. 1992) Children are encouraged to talk and write about how they solve problems. Charts and graphs are presented as important forms of mathematical representation. However, the notion of communication is often viewed solely from the perspective of a verbal linguistic medium - talking about math. Connections between mathematical forms of representation and their role as a means of communication are often neglected. The *Investigations in Number, Data, and Space* elementary mathematics curriculum developed by TERC goes some way towards developing explicit understanding of mathematics as a system of representation and notation, particularly in the emphasis on children's development of their own organization and notations for problems and problem solving activities. However, we know of no other material that uses communication as an activity in itself to encourage children to evaluate and understand representational choices.

The theme of coding and secret messages has been used in other mathematics learning software for middle school aged children. Geared to appeal to older children than *KidCode's* target audience, these software applications use the coding theme to develop particular types of mathematical skills with data analysis, pattern recognition and linear functions.

The Middle School Mathematics through Applications Project (MMAP), includes a Codes Inc. unit in which students act as cryptographers to design and analyze codes. (Goldman and Moschkovich, 1995, 1997) The software includes a set of tools which help students work with patterns, explore algebraic functions, and use tables, matrices, verbal rules and graphs to represent number patterns. As they work with these tools to design and break codes, students gain familiarity with many topics in the middle school mathematics curriculum. The MMAP software environments are specifically intended to support and encourage conversations between students. However, they differ considerably from *KidCode* in that they are designed to fit the cooperative problem solving model described earlier. The coding activities are not intended to be used to effectuate communication among the students.

Top Secret Decoder[1] is commercial software intended for education and entertainment that uses the theme of coding and decoding. Like the MMAP software, *Top Secret Decoder* is most appropriate for middle school aged children. The software enables children to type a message and have the computer automatically code the message with any code selected from a set of approximately twenty codes included with the program.

The software includes a Challenge Master mode in which the computer generates a coded message and the child is given the task of decoding the message. The decoding game is designed to appeal to children who play by themselves but the software is not really intended to support cooperative learning. Children can use the software to generate a coded message which can be printed and delivered by hand but it does not include electronic interaction. The software may help children improve their understanding of figurative language and pattern analysis. However, the fact that the computer always performs the encoding and that representational choice is only superficially under the control of the user, limits its efficacy in encouraging understanding of representation as it relates to communication or to mathematics.

Formative Evaluation Procedure

Subjects participated in pairs in each testing session. Pairs were usually made up of children from the same grade. A few sessions consisted of a child who played with an adult or a child from a higher grade as partner. Most testing sessions involved multiple games and lasted between 1 and 2 hours. It was not possible to test all four games in a single session. Therefore, we usually had the children participate in at least two separate sessions so that we could examine their reaction to the entire sequence of games. In most cases, children were tested at school during school hours in sessions that resembled one-on-one remediation. The children interacted with a developmental psychologist or mathematics learning specialist while a second expert attended as an observer.

Pairs of subjects were told that they would be playing a series of mathematics games that involved passing secret messages between each other. One player would make a secret message, and the other player would have to figure out what the message means. Throughout each game, subjects sat opposite each other at a table with a cardboard divider between them so that they could interact with the game materials without having their partner see what they were doing.

Each game was administered following the same general procedure. First, subjects were instructed how to play each game by talking the subjects through an

[1] *Top Secret Decoder* is a trademark of Harrison Fox & Company.

Figure 1. Sample Rebus Game Materials.

Message to be coded.

Message under construction.

Board of symbols to use.

example version of the game, giving them step-by-step instructions for how to code and decode a message, and in some instances, for how to use that message to solve a problem. Second, subjects played each game in multiple versions. Multiple versions were used to test different game materials, and to give subjects the opportunity to take on the roles of both coder and decoder. Complete sessions including instructions, subject interactions and game choices were documented through video and observational notes. After each game was completed, subjects were asked how they liked the game. They were also asked to explain specific reactions, behaviors, and game choices.

Rebus Game: Introduction to using symbols.

In this game the coder was given a sentence with some of the words underlined. He or she was also presented with a second copy of the same message in which the underlined words were absent. In their place was a dot of velcro. The coder choose symbols or pictures to represent the underlined words and put the selected symbol on the velcro in place of the underlined word. The completed message which consisted of words and symbol representations was then passed to the decoder. The decoder's job was to reconstruct the original sentence from the rebus style document transmitted by the coder. (See Figure 1. above.)

We used the Rebus game to introduce the children to the structure of the games. In playing this game the children learned that the overall objective was to get their partner to understand the message. They learned what it meant to play the role of coder and of decoder, and began to understand how to anticipate (or infer) the thinking of their partner.

Other important aspects of the Rebus game that foreshadowed important features of later, more obviously mathematical games were recognition that:

(a) objects, events, ideas and properties of objects can be represented by symbols;
(b) an important purpose of symbols help us to communicate with other people;
(c) symbols can have many different forms and that a given symbol can have many meanings;
(d) symbol choice is at the discretion of the speaker. Some symbols are better than others because they are more easily interpreted by the recipient. Abstraction in symbol choice enhances predictable communication by restricting possible interpretations; and
(e) to encourage participants to cast off attempts to "get the correct answer". To show children that the point of language is communication. That the interchange and multiple iterations are

expected. That mistakes are illuminating and fun.

A wide variety of symbols and pictures were available to the children to replace the underlined words in the sentences above. Five classes of representations were used - (1) artistic drawings, (2) photographs, (3) schematic drawings, (4) phonetic spellings, and (5) rebus-style spellings. Approximately 40 different symbols were available and we tried alternative methods for arranging the symbols so that the children could more easily find a symbol to represent a given word. Sample sentences included:

- When I see the monsters I run to my house.
- I see ten girls dancing in the rain.
- What has a head, a tail, and no body? A penny.
- The clown said, "Why is your nose not twelve inches?" Because then it would be a foot.

The Rebus game was played repeatedly with four or five different sentences. Each subject coded and decoded at least 2 sentences. Two different ways of presenting the symbols to the subjects were tested. One method involved presenting subjects with large boards that included symbols for different words which belonged to higher level categories (e.g., symbols for animals, plants, girls, and people were included on one board labeled, *living things)*. The other method simply involved the presentation of symbols with the words they were meant to represent (e.g., along with the word girls subjects were presented with artistic, photographic, schematic, phonetic, and rebus-like representations of girls).

All children were able to play the game and to use the materials independently after being guided through one or more rounds. Younger children (pre-first graders) required more help than older children, especially with reading, locating the category from which to select a symbol, and keeping track of decoding errors. In terms of their interactions with the game materials, subjects showed a preference for using visual symbols (schematic, artistic, or photographic representations) over textual symbols (rebus and phonetic representations). Older children sometimes chose textual symbols, especially if visual materials did not work out. Younger children did not use textual symbols at all and had difficulties decoding them.

Subjects made very few errors in coding. A few subjects miscounted when selecting a picture of a number of identical objects to represent a numeric quantity. Younger subjects sometimes had difficulties finding symbols when choices were presented as part of higher level categories. The alternative method of presenting symbol choices word by word was more manageable for the younger children.

The most consistent decoding errors occurred with symbols that were selected to represent a number (e.g., a picture of 10 bananas for the number 10). Subjects tended to interpret these symbols as representing the objects depicted (i.e., bananas) as opposed to one of their qualities (i.e., the number 10). Interestingly, difficulties in decoding numbers as represented by a set of objects was universal for all subjects and did not have any relationship to age. Adults who played the games had just as much trouble decoding this representation as did the youngest children. When a partner misinterpreted a symbol representing a number, older children and adults were more likely to use an alphanumeric representation when recoding.

Implications for Collaborative Learning

We found that the context of interactive communication is very appealing to the children we worked with. An unanticipated result is that the games help children and adults to get over the habit of judging decisions and constructions as right or wrong. In these games, there is usually not a correct method for coding a message. Coding errors can occur but these are of secondary importance and will generally be recognized by software which prompts a coder to correct them before a message is sent. What matters is that the recipient can understand the message. Math becomes less a matter of right/wrong and instead a question of using symbols and learning to translate. It becomes less a matter of applying the rules and more a matter of finding the meaning.

Often in the Rebus game, children found it necessary to recode a message three or four times before the recipient understood. In these cases, players quickly realized that although all codings were equally valid, their partner would not necessarily understand a particular coding and at times could only get the meaning after a sequence of codings. All players, adults and well as children, began the games with a test taking attitude in which they were determined to "get the right answer". After one or two rounds of the Rebus game, they all switched objectives and began to focus on the anticipated reaction of their partner to the coded messages. We feel that the attitude shift and the focus on communication is a very important feature of the games - particularly as it applies to understanding mathematics as a language.

Another surprise was that gender correlated to player interpretations of game objectives. We assumed all children would accept the cooperative nature of the games and try to find codings to help one's partner decipher a message. However, some boys choose to play competitively by coding messages so that they would be difficult for a partner to decipher. In these boys' implicit interpretation, a game could be "won" by decoding messages more accurately or more quickly than one's partner. It was interesting that boys who took a competitive stance, did not stray from appropriate codings even though we did not provide guidelines or suggest that there may be a difference between appropriate and inappropriate codings. As a result of the self-imposed guidelines that competitive players appeared to devise, the games could still be played and remained viable for all of the children we worked with. Nevertheless, because we find the games an excellent setting to encourage cooperative play, we are examining how the software can incorporate incentives such as team play, a scoring method with a roster of high achieving partnerships, or specialized introductions to the games, to encourage all players to adopt a cooperative approach.

Implications for Software Design

Formative evaluation and iterative user interface design have become essential methodologies for software developers (Neilsen, 1993). Using paper prototypes (Rettig, 1994), we were able to rapidly explore and redesign many ideas for user interfaces and for cooperative game design. Many of the games and game materials were altered significantly as a result of watching children use the paper materials. Some findings and design implications specific to the Rebus game were:

(a) a few symbols were incomprehensible to many of the children => find new symbols.
(b) most younger children did not use alphanumeric representations => possible tailoring of the game to grade level.
(c) large numbers of symbol choices were confusing and unnecessary, => symbol libraries should be moderate sized and possibly tailored by grade level.
(d) during the search for a symbol replacement children often got caught up in examining images and forgot the word that they were trying to replace. => keep message to be coded and choices from the symbol library visible on the same screen during coding.
(e) organization of symbols by category was very confusing to younger children. Organization by words was helpful to them.
(f) older children were able to, and very much enjoyed, making up their own messages to be coded. After having become familiar with the symbols in the symbol library, they readily created messages for which appropriate symbols were available.
(g) younger children required text to speech capability but had no problem playing the games otherwise.

One of the most important findings from the evaluation was that careful instruction and/or other scaffolding is required in order for the children to understand and play the games. Children were initially clearly confused with the objective and procedure for many of the games. It often took one or two rounds of tentative play before they became comfortable and involved in the games. Often the children who seemed most confused initially were the ones who became most intensely involved later. The difficulty that children had grasping the game may have had something to do with our initial instructions. However, all children needed at least one walk through of each game and the number of rounds required decreased with age. We believe that the structure of the activity was unfamiliar to them and that the degree of confusion was related to their level of symbolic processing development. The observation that, after a small number of sample sessions, all of the children could play most of the games independently (assuming textual material was read to non-readers) and enjoyed them very much suggests a developmental readiness across the entire age range.

These findings have important implications for how the computer software should be designed. It is clear that the provision of adequate instruction within the context of the games will be a critical factor that determines the success of the program. Since, the games are intended to be "self-teaching" it was important to find ways to describe game play to the children. We found that some forms of instruction are much more effective than others. In particular, we found that walking through an example of game play was far better than verbal directions for how to play. Clearly it is important to provide demo walk throughs of game play and scaffolding so that games are presented in a sequence of gradually increasing complexity and abstraction.

Although the Rebus game was enjoyed by all children regardless of age or level of development, we found that developmental readiness was an important determinant of the children's level of interest for some of the games. Younger children were particularly fascinated with a mathematics coding game that resembled the game of Battleship. Older children, who tend to have more exposure to symbolic language, were more readily able to utilize the coding models and had less trouble with more abstract games and games that required greater sophistication of geometric reasoning. Younger children needed more trials and more systematic instruction. However, once they grasped the coding model used in a game, they were easily able to use what they learned and apply it to new situations. In fact, we were surprised that even children who did not yet read could play the games if an adult read textual messages.

Conclusion

We found that the *KidCode* games improve children's skills with particular representations used commonly in mathematics and they seem to improve the children's competence with symbolic processing. While all children who played the games in sequence were able to understand and play each game independently after being guided through one or two examples, children had a great deal of trouble when the games were not presented in sequence and games with higher levels of coding abstraction were played without the benefit of experience with earlier games.

It was our further aim that the games develop children's conscious understanding of symbolic representation and confidence in their ability to translate any representation to retrieve it's underlying meaning. We believe that experience with various kinds of representations will give children a foundation so that they are not intimidated when presented with a complex looking mathematical formula and so that they consciously experiment with alternative representations when asked to solve a mathematics problem. Only a more extensive longitudinal study that includes children's teachers could determine the full impact of *KidCode* on more general mathematics learning.

Paper prototypes used in this work have resulted in substantial improvements in the design of the KidCode software at very early stages of development. Because we were able to revise quickly and make significant alterations in our initial conceptions, we were able to completely revise many of the games in response to children's interactions with one another and the materials. Moreover, we were able to evaluate and construct the sequence of games to provide good scaffolding to support the children's developing understanding.

We also found that the interaction model of electronic mail works well to structure a series of interactions as a two person game involving a coder and decoder. Occasionally we cautioned a player not to try to peek at a partner's work in progress, but found that the children quickly adapted to the nature of the communication as turn taking that yielded new information with each turn. We also discovered that a few important questions can only be resolved by evaluation of the software. Not surprisingly, most of these questions pertain to the nature of computer mediated interaction. Working with paper materials, children sat together in a room and could monitor one another's reactions in a direct way. To sustain this level of engagement for young children we may wish to incorporate features not normally associated with electronic mail such as audio or video annotations. The effectiveness of the online help system as a sole source of instruction and the degree to which the media supports engaging interaction will guide our research and software development as we proceed in the next phase of this work.

Selected References

L.J. Bricker, Tanimoto, S.L. Rothenberg, A.I., Hutama, D.C. and Wong, T.H. Multiplayer Activities that Develop Mathematical Coordination. *CSCL '95 Proceedings*, Lawrence Erlbaum Associates, Inc. New Jersey. 1995.

T.P.Carpenter, Ansell, E., Franke, M. L., Fennema, E., & Weisbeck, L. Models of problem-solving: A study of kindergarten children's problem solving processes. *Journal for Research in Mathematics, 24(5),* 1993. pp. 428-441.

J.S. DeLoache. Rapid change in the symbolic functioning of very young children. *Science, 238*, 1987. pp 1556-1557.

D.C. Edelson, O'Neill, D.K., Gomez, L.M. and D'Amico, L. A Design for Effective Support of Inquiry and Collaboration. *CSCL '95 Proceedings*, Lawrence Erlbaum Associates, Inc. New Jersey. 1995.

H. P. Ginsburg, *Children's Arithmetic*. (Second edition.), PRO-ED, Inc., Austin, TX, 1989.

H. P. Ginsburg. Mathematics Learning Disabilities: A View From Developmental Psychology. *Journal of Learning Disabilities, 30(1),* 1997. pp. *20-33*.

S. R. Goldman, Hasselbring, Ted S. and The Cognition and Technology Group at Vanderbilt. Achieving Meaningful Mathematics Literacy for Students with Learning Disabilities. *Journal of Learning Disabilities, 30(2),* 1997. pp. 198-208.

S. Goldman and J.N. Moschkovich. Environments for Collaborating Mathematically. *CSCL '95 Proceedings*, Lawrence Erlbaum Associates, Inc. New Jersey. 1995.

S. Goldman and J.N. Moschkovich et al. MMAP Web site. http://www.irl.org/mmap/MMAPinfo.html

B. Greer. The Two Faces of Mathematics. *Journal of Structural Learning,* Vol. 10, 1989. pp 79-82.

C. Gutwin, Stark, G. and Greenberg, S. Support for Workspace Awareness in Educational Groupware. *CSCL '95 Proceedings*, Lawrence Erlbaum Associates, Inc. New Jersey. 1995.

M. Guzdial, Turns, J., Rappin T. and Carlson D. Collaborative Support for Learning in Complex Domains. *CSCL '95 Proceedings*, Lawrence Erlbaum Associates, Inc. New Jersey. 1995.

Harrison Fox & Co., Nanny M., and Mohl, B. *Top Secret Decoder*. Houghton Mifflin Interactive. 1996.

National Council of Teachers of Mathematics. *Curriculum and Evaluation Standards for School Mathematics*, 1989.

J. Nielsen. Iterative User-Interface Design. *Computer*, Vol. 26, No. 11, Nov. 1993. pp 32-41.

M. Rettig. Prototyping for tiny fingers. *Communications of the ACM*, Vol. 37, No. 4, April 1994. pp 21-27.

D. Scott, B. Williams, and K. Hyslip. Mathematics as communication: journal writing in 2nd grade. *Childhood Education*, Vol. 69, No. 1, Fall 1992. pp 15-18.

TERC (Technical Education Resource Centers). *Investigations in Number, Data, and Space*. Dale Seymour Publications. Palo Alto, CA. 1997.