Bruce, Bertram C. (1987). An examination of the role of computers in teaching language and literature. In James R. Squire (Ed.), *The dynamics of language learning: Research in reading and English* (pp. 277-293). Urbana, IL: National Council of Teachers of English.

Computers and Language: A Look to the Future

Bertram C. Bruce

Bolt Beranek and Newman Inc.
10 Moulton Street
Cambridge, Massachusetts 02238

December 1985

Abstract

Computers and language are intimately connected in four ways: (1) the computer is a tool for representing knowledge through symbols, (2) it is a device for interpreting symbolic structures, (3) it is a communication device, and (4) it is a redefinable tool. This essay considers these four aspects of computers by taking an excursion into a classroom of the year 2010, and then looking back to evidence of precursors in classrooms of the 1980's.

What place should the computer have in the language arts classroom? [I use the term "language arts" in a broad sense to encompass classrooms at any grade level in which the focus is on learning how to use, understand, and appreciate language.] Many people would say "none at all." If they see any connection between computers and language learning it is that the study of language, with its attendant emphasis on culture and history, and especially, the study of literature, should serve as an antidote to a society that seems increasingly centered on technology.

Of all the new technologies, the one which appears to threaten humanistic learning and values the most is the computer. Thus, it seems appropriate to focus a discussion of technology in education on computers. But there are deeper reasons for focussing in this way which relate to the fundamental nature of computers. First, the computer is a tool for representing knowledge through symbols: as such, the essence of computer use is identical to what we do when we use language. Second, the computer is a device for interpreting symbolic structures, for making sense of linguistic representations. Third, the computer is a communication device. It can store representations of information, but more importantly, can transmit these representations to other people and other communication devices. Finally, the computer is an object in the process of becoming. Like other tools, the computer can be used in a variety of ways; unlike the others, its very nature is to be redefinable.

These aspects of the computer are not assumed in many of the discussions of computers in their relation to language arts, discussions on issues such as video games versus reading, the elevation of science and technology over the humanities, and methods of or appropriateness of computer-assisted instruction. By not addressing the deeper aspects of computers, we foster an either/or atmosphere in which the language arts are often denigrated. Worse still, we fail to assert control over the direction of a tool which has an unquestionably powerful potential for teaching about language.

In the next section, I discuss these four aspects of the computer's relation to language. These aspects derive not just from consideration of computer applications, but rather, from an analysis of the computer's essential functions. Following that I describe a classroom of the future, one which is only a slightly extended composite of today's classrooms. For each aspect of the future classroom, I have tried to identify some current activities that capture at least some of its potential. One purpose of this excursion into the future is to demonstrate that it is appropriate to discuss computer use when thinking of language. The case becomes, then, not that computers are good or bad for teaching language, but rather that they inherently belong in that province, and should be shaped by the people who live there. The last section raises some questions for research based on this thesis.

Computers As Language Machines

We tend to think of the computer, quite naturally, as a device that computes, in particular, as one that essentially adds numbers very fast. In every field in which computers have been used, including the military, industry, business, mathematics, medicine, science, social science, the humanities, and education, the computer was perceived first as a device for counting and carrying out simple mathematical operations. Thus, the military used the ENIAC for calculating ballistics trajectories; businesses used early office machines for keeping accounts; medical researchers collected statistical data on correlations of symptoms and diseases; humanists used computer word counts for authorship studies; and educators put computers in schools to teach arithmetic.

Today, people in each of these fields are beginning to use computers in quite different ways; specifically, they are using them for help in writing and reading, to carry out symbolic transformations, and to communicate with other people. These new uses are not merely additions to the computer's repertoire, but rather, precursors of the computer's fundamental role as the general

language machine; or to use Steven Jobs's phrase, "wheels for the mind."

Why do we continue the pattern of using computers for numbers first and words second? Perhaps we have failed to understand some of the subtle relations between computers and language. There are four of these I would like to discuss here--the computer as a means for representing knowledge, as a device for interpreting symbols, as a communication device, and as a redefinable tool.

Computers are tools for representing knowledge

A computer is, at its core, not just a collection of flip-flops or integrated circuits. Nor is it simply a big numerical calculator. At the deepest level, a computer is a device for encoding and storing symbols. Symbols thus encoded can be associated with other symbols; in that way, symbolic structures of arbitrary complexity can be constructed and maintained. Thus, the computer is a tool for representing any knowledge that can symbolized.

Computers are tools for interpreting symbols

Other technologies, for example, the book, are also convenient for recording symbols. But computers differ from books and other technologies in a way which has a special significance for the teaching of language. Computers are physical realizations of the concept of a totally general symbol manipulator, a device which can not only store, but also create, transform, or interpret essentially any symbolic representation. Thus, when we talk of what computers are, or should be, we must operate in the realm of Kant, Frege, or Levi-Strauss, not that of the BASIC programming manual.

Computers are communication devices

Computers are also communication devices: They can store and interpret symbols, but they can transmit them as well. The use of computers is transforming every other communication device, from telephones to video discs. In fact, the communications industry, whether its physical medium be books, magnetic tapes, or cathode ray tubes is increasingly dependent upon computers because only computers make possible the control flexible and precise enough to transmit just what is needed, or to record the right data. To a large extent the computer and the communications industries have already become one. The consequences of this fact for language use are significant.

Computers are redefinable tools

There is a fourth reason why computers are intimately tied to language: They are redefinable. Unlike typewriters, tape recorders, ditto machines, telephones, televisions, and other technological devices that might be used in education, the computer is a tool whose very nature is a process. Many tools undergo rapid development, but the computer is itself a tool for making tools. For example, a computer, when unpacked from its box might appear to be a LOGO (Feurzeig & Papert, 1969; Feurzeig, Papert, Bloom, Grant, & Solomon, 1969) (or some other programming language) machine. That is, one could use it to carry out the basic LOGO functions, such as moving a "turtle" about the screen. But one could also use LOGO, or any general purpose computer language, to define new functions, for instance, a program to find rhymes in a dictionary. The added functions would mean

that one's machine would no longer be simply a LOGO machine, but rather, a LOGO-PLUS machine. One could also turn the LOGO machine into a BASIC machine, by writing the proper function (an "interpreter"). In fact, there is no known theoretical limit to what sort of machine the computer could become.

The protean nature of the computer implies that we always need to look beyond current uses of computers in order to assess whether and how they might best be used. In particular, we need to consider functions other than the usual ones of classroom management, multiple choice testing, drill and practice and frame-based computer assisted instruction. Most importantly, we need to explore computers as general symbol manipulators. The next section is designed to encourage some speculation regarding desirable functions for computers.

The Language Classroom of 2010

This section presents some sketches of how computers have been and might be used in teaching reading and writing. The first sketch focuses on the computer as a tool for knowledge representation. The second emphasizes the computer's role as interpreter of symbols. The third looks at the computer as a communication device--for reading and sharing ideas, for collaborative writing, and for networking. The last sketch looks at the computer's redefinability and the implications for creativity. For each sketch we will look in on Hannah Lerner and her classroom in the year 2010, then look backward to the 1980's to find precursors of what we see in her class.

Knowledge Representation

When students in Hannah Lerner's class in the year 2010 work at the computer, they engage in what they call "idea processing." Idea processing means working at the level of concepts and higher level text structures, such as "counterargument" or "elaboration." When students process ideas with the computer, they think of what they do as building structures, testing, and debugging. Thus, idea processing goes far beyond the word processing familiar in the 1980's. Similarly, the students might be said to be programming, but again, the activity bears only a slight resemblance to the old rigid procedural paradigms. The focus is on the project they are doing, not on the syntactic details of either a programming language or a word processor. What has happened in 2010 is a merging of two earlier modes of computational interaction. Computer programming per se has begun to resemble natural language use and writing with the aid of a machine has come to resemble very high level programming

The reason for this is that defining a procedure for a computer to carry out or creating a text each requires the person to formulate and organize ideas. Writers of programs and texts are both concerned with planning and revision; they both need to be aware of their audience (Newkirk, 1985). With programming sufficiently removed from the bits and bytes level and text processing from the letter by letter level, these two once disparate activities become essentially one. As a result, Hannah's students often find themselves using the computer to wrestle with ideas in the same way, regardless of the end product--a text, a computer program, a graphical display, or simply a deeper understanding

of a domain of study.

Precursors of the trend could have been seen in the 70's and 80's. For example, programming languages such as SMALLTALK (Goldberg & Robson, 1983) allowed a programmer to define an <u>object</u> and a set of rules for how that object should behave (how to display itself on a CRT screen, how to provide information about its current status, how to change as a result of changes in its environment, etc.) This tended to free the programmer from concern about the precise sequence of actions the computer should take. Similarly, <u>rule based systems</u> such as MYCIN (Davis, Buchanan, & Shortliffe, 1977) allowed the programmer to define hundreds of rules of the form IF X THEN Y without concern for which rule should be checked first. @footnote(In the case of MYCIN the set of rules could be activated by a patient's history to help a physician diagnose a bacterial infection.)

While object oriented languages and rule based systems were being developed artificial intelligence programmers were also developing higher level functions in their programming languages. For example, transition networks (Woods, 1970) were developed as a language for describing in computational terms the set of grammatical rules for a language. Each such language enhancement moved programmers further from the machine qua machine and closer to the problems they were addressing.

At the same time, word processors were giving way to idea processors (see Olds, 1985). The early signs of this change could be seen in the emergence of programs to help with planning a text (PLANNER in QUILL, Bruce & Rubin, 1984), organizing ideas (THINKTANK, Owens, 1984), examining texts in a non-linear fashion (ORG in WRITER'S WORKBENCH, MacDonald, 1983), managing text annotations (ANNOLAND in Authoringland, Brown, 1983), and exploring and modifying data bases. As this class of programs matured it enabled a form of interaction between a person and an emerging text in which the linking of ideas, the examination of an argument, or the search for related concepts was as easy as the correction of a spelling error with a word processor.

For example, Linda Juliano, a sixth grade teacher in Cambridge, Massachusetts in 1984, wanted to push the limits of how a computer might facilitate language use. One of her students had written a story about a trip to the circus which was extremely long and unfocused. The student didn't know how to cope with revising the text, to some extent because of the volume of material. The text had been written using a text editor known as Writer's Assistant (Levin, Boruta, & Vasconcellos, 1982), which had a special feature (called "Mix") that allowed a writer to start every sentence at the left margin. Ordinarily, this was used to check for syntactic errors -- first letter capitalization, end punctuation, repetitive first words, and so on. Linda saw that it could also be used to facilitate examining and manipulating a long text. She suggested the student format his text in the separated sentence fashion, print it out, and cut the sentences apart with scissors (a pre-2010 device used by writers to help in revising). With the sentences apart, it was easy to experiment with various deletions and rearrangements. Once the student had formed his revised text as a pile of sentences he used the text editor again to re-create the final text. The computer thus became a tool for thinking of his text in a new way.

<u>Interpretation</u> *Although Hannah continues to be the essential teacher of her class, the computer plays an important role as assistant tutor. This is possible because the computer interprets, not just*

represents, symbols. For example, the computer can analyze stylistic features of the text, everything from spelling to paragraph forms, and provide information for the writer to use in revising.

The computer can also model processes of revision, by showing successive alterations of a text, together with audio or textual annotations giving the reasons the author had for changes. This modeling can be run in slow mode, showing letter by letter changes, or fast-forward, showing higher-level revisions. Since the computer has stored examples from Hannah's own writing and the writing of experts, as well as that of students in the class, the study of various revision strategies often leads to valuable discussion of writing and writing styles.

Back in 1985 a program which took advantage of the computer as a symbol manipulator was ILIAD (Bates, Beinashowitz, Ingria, & Wilson, 1981). ILIAD had a large amount of knowledge about transformational grammar (Chomsky, 1965) that enabled it to generate many different possible transformation of any given sentence (if it knows the parts of speech). For example, the sentence "Bill ate the cake" could be transformed into: "Did Bill eat the cake?", "Bill should eat the cake.", "Didn't he eat it?", or "It might have been eaten." With this capability, a variety of activities could be designed to help children develop the ability to express their ideas in different ways.

Sharples (1980) had developed several programs along this line together with a set of activities that he used to teach writing in a fifth grade classroom. One of these programs was GRAM, which generated text on the basis of a set of rewrite rules, which were expanded until a string of words was generated. For example, Sharples developed a poetry generator by specifying that a poem could be rewritten as a title and a body. The title could be any noun phrase. The body could be any number of lines. He provided several different possible definitions for a line (e.g., noun phrase plus intransitive verb phrase plus preposition plus noun phrase). A noun phrase in turn could be a plural noun, and a plural noun might be "lilies" or "frogs." The poetry generator made each of these choices randomly, thus producing a poem within the constraints of the grammar. By manipulating the grammar, students came to see how different constraints produce different kinds of poems.

Another program, TRAN, allowed students to write their own transformations, like those in ILIAD. These were written as pattern-action rules: If a piece of text matches the pattern on the left side of the rule, that part of the text is replaced by the right hand side of the rule. For example, the rule "noun1 1 noun2 --> noun2 1 noun1" swaps the first two nouns in a sentence (the 1 between noun1 and noun2 allows for a string of any length). Sharples worked out a set of activities based on TRAN to teach children sentence combining and other manipulations of sentences. In one activity children wrote descriptions and the computer replaced all the adjectives it knew by a star. The object for the children was to try to produce as many adjectives as they could that the computer did not know. These activities allowed children to explore language by manipulating the language systematically.

Another symbol manipulating program was WRITER'S WORKBENCH (Frase, 1983; Gingrich, 1983; MacDonald, 1983), an automated Strunk & White (1972). It analyzed a text and made comments that the writer could choose to use or ignore. For example, it could point out frequent use of words like "seem" or the conjunction "and" between clauses. It was originally designed for adults doing technical writing, but was later used as a tool for learning to write.

<u>Communication</u>

Hannah entered her classroom well before her students were expected to arrive. She had found that in the minutes before they appeared she could check her mail on the computer and review plans for the day. On this particular day, one group of students would be completing a botany project they had begun earlier in the spring. Its purpose was to compare bean plant growth rates at various altitudes and under various climatic conditions.

"Good morning, Ms. Lerner."

The untimely end of the quiet period was signaled by the early arrival of two of Hannah's students, Kit and Adam. Kit immediately went to his computer to see if there had been any additions to the plant data base. Luckily, there was a message from Sao Paulo presenting some data from their greenhouse project. These data would be incorporated with other data from Rome, Tokyo, Mexico City, and Hannah's classroom in producing the science group's botany report.

Meanwhile, Adam sat down at another computer to see what changes his co-authors had made in their collaborative novel project. Using a multicolored screen with holographic projections he could examine both the original text and any author's additions or alterations. New portions of texts could be alternately highlighted or blended into the original. Comments by one author on another's passage could also be examined, or not, as Adam chose. The three dimensional quality of the display conveyed a sense of what texts and comments were available in addition to those immediately visible. Adam was eager to read what his co-authors had done; perhaps one had sent in more text last night. It would be interesting to see if their semantic network for the text, also presentable graphically, had changed because of any text changes.

Hannah's class in the year 2010 is in a sense a group of people who get together in one place and time for learning. But in a larger sense, the boundaries of the class are not easy to define. Students who are away from school because of illness, family business, bad weather, or whatever reason often check in via a network that links together their homes, the school, other schools and the outside world. This network allows transmission of text, pictures and graphics, even audio--voice, music, other sounds, and video. One problem that arises is remembering where someone really is. Since it is as easy to share information with someone at a computer five thousand miles away as with one five feet away, students have to learn to observe carefully the dateline that comes with each message. Networking also diminishes the distance created by time. Lisa can read a story finished on another continent six hours ago while she was sleeping. She can search a data base containing the entire Library of Congress to read texts written in any time and place. The process of searching that data base is similar to the one she goes through in looking for writings of her classmates, since most of the students' writing is stored in a network-accessible data base, too. (Lisa also keeps a journal in a traditional blank book, believing that no single form of technology is appropriate for all types of writing.>

Back in 1982, Jim Aldridge's sixth-grade class in Hartford was also using the computer for learning through reading and writing.@footnote Jim described a special time in the morning before class when he turned on his "electronic classroom." There was a television, used for news and educational programs, a microcomputer, and a table-top greenhouse project with vegetables in pots and

fluorescent lights. During that time, Jim, like Hannah, would often do his own writing, or reading of children's works.

Each of Jim's students had a plant growing in the greenhouse. They would periodically take the plant over to the computer to record data on its growth. They would also take their science texts to the computer to compare diagrams in the text with the actual plant structures. Using questions written by Jim, the computer served as a mediator between the words of the text and the real world of the plant. After collecting data over an extended time, the students could write a lab report detailing their observations.

Meanwhile, five girls in Jim's class were using the computer for the fourth chapter of their romantic novel about Menudo, the Puerto Rican rock group. The novel was inspired by another project in the class, writing a prospectus for a to-be-produced class play. But the Menudo story took a separate course, becoming a secret saga shared among only its authors and a few select friends. The girls would, at every possible moment, add pieces to their collaborative text. Sometimes they would write literally side by side, in groups of two or three at the computer. At other times they would add a portion to be read and perhaps modified their collaborators later.

Unfortunately, these girls had only a text editor for their writing. Text editors facilitate writing because they enable easy editing and help in the production of clear copy. But they facilitate neither collaborative writing, nor thinking of ideas and text in larger units. Authoringland (Brown, 1983; Watt, 1983) is a system, part realized and part envisioned, which does just that. In the Authoringland computer environment a writer can modify a text but leave an "audit trail," which shows other authors (or the original author) what was changed, when and why. A writer can also make comments: passing thoughts, identifications of problems in the text, concepts to be elaborated later, or, comments on other comments. The information in the computer is then no longer a single piece of connected text, but a network of text parts, ideas, reasons for changes, and notes to think about. The computer allows a simple and clear graphic representation of relevant portions of this network, so that the writer can explore it, modify it, or draw from it a writing product.

Early in 1984, students in Shungnak Elementary School in Alaska used a satellite to talk with students in the nearby village of Kiana and the city of Fairbanks. They then used a computer to write, edit, and publish an article in Educational Technology/Alaska about their audio conference:

> We talked to Kiana and Fairbanks to learn more about different communities. To get ready for the conference we wrote letter and took pictures of ourselves, then we sent them to Kiana and Fairbanks.
>
> Two days before the audio conference we wrote our questions on a piece of paper. On the day of the conference the first thing we did was introduce ourselves, then we asked our questions.
>
> We learned a few things from Kiana and Fairbanks. Kiana told us how to make an igloo. . . . . We found out that Kiana eats the same Eskimo food we do. Some of these foods are

frozen fish (quaq), Eskimo ice cream (akutuq) and dried fish (paniqtuq). When one girl in Fairbanks told us her father had a plane and she might come and visit us, we were very excited.

Towards the end of the conference we sang a song to the other schools. The song was Pearly Shells. First we sang it in English and then we sang it in Inupiaq. . . . . We enjoyed talking to the kids in the other communities. We discovered we have many things in common, but also some of us do things differently.

While these students were learning about others through audio conferencing, reading, and writing, students in other towns were also using networking to communicate. Some of these students used CCNN, the Computer Chronicles News Network (Riel, 1983), a UPI or AP for kids. Members of the network wrote stories, poems, editorials, and other articles appropriate for a newspaper and sent them via a computer network to a large computer in Virginia. When a class wanted to produce a newspaper or magazine, they could then supplement their own articles with selected articles from CCNN. Naturally, in order to make a selection, they had to read a large number of articles others had written; in writing they had to think of their audience, taking into account the fact that their readers had different cultural experiences and background knowledge.

The computer was doing several things to facilitate the sharing of writing seen with CCNN. First, an article was transmitted almost immediately to anyone who wanted to read it. Second, there was essentially no limit to the number of possible readers. Moreover, the author did not have to make multiple hard copies and address envelopes to all the readers. Third, if a reader wished to incorporate a CCNN article into his or her newspaper, the text was already in a machine readable form so it could be formatted, edited, and merged with other newspaper articles. Some examples of CCNN articles are included in the Appendix to this paper.

Redefinition

Back in 1984, Nancy Sopp's (1984) junior high students in Fairbanks, Alaska, wanted to write a story in the form of a computer Adventure game (Addams, 1985). This would be an interactive text in which the next passage a reader sees depends upon his or her actions. They realized that to do so it would help to have a program to handle the details of connecting reader actions to text passages so that they could focus on the texts per se. Moreover, this program should be suitable for any set of texts, not just the first draft they would write. What they did was to write an Adventure game maker using the language LOGO. The result was a new language, both more powerful, and more specialized. Their project already blurred the traditional boundaries between learning about computers and learning about language.

A generalization of the Adventure game maker was a computer language called ITI (Interactive Text Interpreter, Levin, 1982). ITI was a "high level" language that redefined what the comptuer could do. Using it, students or teachers could create poetry generators, STORYMAKER-like programs (Rubin, 1982) or Adventure games. The sports editor for a student newspaper, for example, could create a tool to use in writing sports stories that would remind a writer to include the final score or to conform to stylistic conventions. Levin and his colleagues used ITI to create tools such as an Expository Writing

Tool; a Letter Writer, which helped students learn various formats for letters; a Narrative Writing Tool; a Poetry Prompter; and Computer Chronicles, a tool for newspaper writing. These tools showed how the computer could be successfully redefined, first as a PASCAL machine, then as an ITI machine, and finally as, for instance, an Expository Writing Tool.

Future Research

If it is true that the computer is the general language machine, then those interested in language learning might reasonably be expected to engage in studies of the computer vis-a-vis language. But the possible connections are many. What are the areas that need the most emphasis?

One area concerns the computer in its knowledge representation function. Today we typically use a computer as a means for representing linear texts. Thus, we can change the spelling of a word, insert a sentence, or delete a paragraph. More complex manipulations of the text tend to detract from a focus on language use. Yet software can be designed to facilitate all sorts of non-linear representations: outlines, associative networks, multiple connections, annotations, and so on. How to design and how to make good use of such possibilities are questions that need much attention.

A second area revolves around the computer as an interpreter of symbolic structures. Here, more work needs to be done on the computer as tutor.@footnote[Taylor (1980) suggests that we think of computer as tutor, tool, or tutee. In the tutor role, the computer teaches directly; in the tool role it assists in doing something, for instance, reading and writing; and in the tutee role it is used as a device that can be "taught" (or redefined) to become something new.] All too often, ideas for the computer as tutor degenerate into constricted and boring activities that diminish rather than enhance students' excitement about language. Nevertheless, the computer has a strong potential as an intelligent tutor for language learning (see Collins, 1985). The computer can present problems, act as a coach, or model the revision process. These approaches need to be explored, especially in conjunction with new uses of the computer as a tool.

A third area of needed research is in the further extension of the computer as a tool for communication. For example, the Alaska QUILL project (Barnhardt, 1984; Bruce & Rubin, 1984b) has begun to look at networking among teachers, which is potentially even more significant than the networking among students (as with CCNN). Moreover, we need to learn whether and how students' active use of language translates to more critical reading. Finally, there needs to be more work on integrating language software with software and activities in other domains, e.g., in science and social science.

A fourth area in which research is needed is on the redefineable nature of computers. This is a powerful concept that may alter our understanding of what language is, or can be. Smith (1982) has argued that the core problems of computer science are not merely analogous to, but identical with, those in the philosophy of language. It is no accident that terms such as "self reference," "interpretation," "syntax," "semantics," "model," or "reflection," appear in discussions of computer languages and architectures. The notion of redefinability, or definability from within, is central to both computer science and language. Moreover, at the level of use, the very act of programming, or (re-)defining, is not unlike the act of writing, with similar ideas of hierarchy, problem solving, and

elegance (see Newkirk, 1985). These relations need to be better understood, as well as applied in developing useful computers.

Finally, this paper has said little about the larger context of use of computers, or of the problems that come with such use. There needs to be more work on equity of access, in terms of hardware, software and the way computers are being used (Michaels, Cazden, & Bruce, 1985; Sanders, 1985). We also need to question both the reasons why schools choose to use computers and the alternatives they forego in doing so. The resources necessary to supply schools with hardware, software, and training cannot be ignored. But, the dollars spent on computers become insignificant against either the rosiest or gloomiest views of how using computers may alter our relationship to language and the world. Will children no longer distinguish the model from the reality, as Weizenbaum (1982) asks, or will the use of models deepen their understanding? Will our sense of what language is diminish or expand as we adopt computer metaphors for our own thinking and communicating (Young, 1984)? Does the ease of revision mean that written texts lose the sense of permanence they once held? What are the consequences of that for society in general? (I am reminded of Kundera, 1981, concern about the "forgetting" of truth in history.) What are the consequences for language learning? Questions such as these need to be investigated thoroughly.

Conclusion

Computers are fundamentally devices for carrying out essential language functions such as creating, interpreting, and communcating symbolic structures. Furthermore, their capabilities are redefinible, or open-ended, in much the same way that language itself is open-ended. Thus, on a theoretical, as well as a practical basis, computers are intimately linked to language.

There are dangers inherent in the use of computers for education; there are also great potential benefits from their use. But assessing the likely effect of computers in education is not a simple matter of comparing lists of pros and cons. One reason is that we simply don't yet know what computers are or could be. What seems clear, though, is that we have underestimated the deep relations between computation and language both at the theoretical and the practical levels. If we are to make the best use of computers for language education we need to ensure that those already involved in that area begin to think more about what computers can and should be.

References

Addams, S. (March 1985). Interactive fiction. Popular Computing, 97-99, 180-182.

Bates, M., Beinashowitz, J., Ingria, R., & Wilson, K. (1981). Generative tutorial systems. Paper presented at the Association for the Development of Computer-based Instructional Systems.

Bates, M. & Wilson, K. (1982). ILIAD: Interactive Language Instruction Assistance for the Deaf - Final Report No. 4771. Bolt Beranek and Newman Inc.

Brown, J. S. (1983). Process versus product: A perspective on tools for communal and informal electronic learning. In S. Newman and E. Poor (Eds.), <u>Education in the electronic age</u>. New York: Learning Lab, WNET/Thirteen Educational Division, 1983.

Bruce, B., & Rubin, A. (1984a). <u>The utilization of technology in the development of basic skills instruction: Written communications</u>. (Report No. 5766). Cambridge, MA: Bolt Beranek and Newman Inc.

Bruce, B., & Rubin, A. (October 1984b). <u>The Quill writing project for Alaska, Final Report</u>. (BBN Report No. 5789). Cambridge, MA: Bolt Beranek and Newman Inc.

Chomsky, N. (1965). <u>Aspects of the theory of syntax</u>. Cambridge, MA: MIT Press.

Church, A. (1932). A set of postulates for the foundation of logic. <u>Annals of Mathematics</u>, <u>33</u>, 346-366.

Collins, A. (1985). Teaching reading and writing with personal computers. In J. Orasanu (Ed.), <u>A decade of readng research: Implications for practice</u>. Hillsdale, N.J.: Erlbaum, 1985. Davis, R., Buchanan, B., & Shortliffe, E. (1977). Production rules as a representation for a knowledge-based consultation program. <u>Artificial Intelligence</u>, <u>8</u>.

Feurzeig, W., & Papert, S. Programming-languages as a conceptual framework for teaching mathematics. <u>Proceedings of the colloquium: Programmed Learning Research: Major Trends.</u> Paris: Dunod, 1969.

Feurzeig, W., Papert, S., Bloom, M., Grant, R., & Solomon, C. <u>Programming language as a conceptual framework for teaching mathematics</u> (Report No. 1969). Cambridge, MA: Bolt Beranek and Newman, 1969.

Flower, L., & Hayes, J. R. (1981). Plans that guide the composing process. In C. M. Frederiksen and J. F. Dominic (Eds.), <u>Writing: The nature, development, and teaching of written communication/Volume 2 Writing: Process, development and communication.</u> Hillsdale, N.J.: Lawrence Erlbaum Associates.

Frase, L. T. (1983). The UNIX Writer's Workbench Software: Philosophy. <u>The Bell System Technical Journal</u>, <u>62</u>, 1883-1890.

Goldberg, A. & Robson, D. (1983). <u>SMALLTALK-80</u>: <u>The language and its implementation</u>. Reading, Mass.: Addison-Wesley Publishing Co.

Gingrich, P. S. (1983). The UNIX Writer's Workbench Software: Results of a field study. <u>The Bell System Technical Journal</u>, <u>62</u>, 1909-1921.

Kundera, M. (1981). <u>The book of laughter and forgetting</u>. New York: Penguin Books.

Levin, J. A. (1982). Microcomputers as interactive communication media: An interactive text interpreter. <u>The Quarterly Newsletter of the Laboratory of Comparative Human Cognition</u>, <u>4</u>, 34-36.

Levin, J. A., Boruta, M. J., & Vasconcellos, M. T. (1982). Microcomputer-based environments for writing: A Writer's Assistant. In A. C. Wilkinson (Ed.), <u>Classroom computers and cognitive science</u>. New York: Academic Press.

Levin, J. A., & Kareev, Y. (1980). Personal computers and education: The challenge to schools. (CHIP Report 98). Center for Human Information Processing. University of California, San Diego.

MacDonald, N. H. (1983). The UNIX Writer's Workbench software: Rationale and design. <u>The Bell System Technical Journal</u>, <u>62</u>, 1891-1908.

Michaels, S., Cazden, C., & Bruce, B. (1985) Whose computer is it anyway? <u>Science for the People,</u> <u>17</u>, 36, 43-44.

Newkirk, T. (1985). Writing and programming: Two modes of composing. <u>Computers, Reading and Language Arts</u>, <u>3</u>.

Olds, H. F. (1985, March). A new generation of word processors. <u>Classroom Computer Learning</u>, pp. 22-25.

Olds, H. F., Jr., Schwartz, J. L., & Willie, N. A. (1980). <u>People and computers: Who teaches whom?</u> Newton, MA: Education Development Center.

Owens, P. (1984). Thinktank and Promptdoc. <u>Popular Computing</u>, April 1984, pp. 186-188.

Riel, M. (1983). Education and ecstasy: Computer chronicles of students writing together. <u>The Quarterly Newsletter of the Laboratory of Comparative Human Cognition</u>, <u>5</u>, 59-67.

Rosegrant, T., & Cooper, R. (1963). <u>Talking screen textwriting program manual: A word processing program for children using a microcomputer and a speech synthesizer</u>. Glendale, Arizona: Computing Adventures, Ltd.

Rubin, A., & Bruce, B. (1985). In B. A. Hutson (Ed.), <u>Quill: Reading and Writing with a microcomputer</u> Advances in Reading/Language Research, Vol. 3, pp. 97-117. JAI Press Inc.

Rubin, A. D. (1980). Making stories, Making sense. <u>Language Arts, 57</u>, 285-298.

Sanders, J. S. (1985, April). Making the computer neuter. <u>The Computing Teacher</u>, pp. 23-27.

Sharples, M. (1980). A computer written language lab. DAI Working Paper No. 134. Artificial Intelligence Department, University of Edinburgh, Scotland.

Smith, B. (1982). Reflection and semantics in a procedural language. MIT Laboratory for Computer Science Report MIT-TR-272.

Sopp, N. P. (1984). Advanced language arts applications. In R. V. Dusseldorp (Ed.), <u>Alaska's Challenge: Computers in education--strategies and solutions</u>. Anchorage, AK: Alaska Association for Computers in Education (P.O. Box 4-562).

Strunk, W., Jr., & White, E. B. (1972). <u>The elements of style</u>. (Revised edition), New York: Macmillan.

Taylor, R. P. (1980). <u>The computer in the school: Tutor, tool, tutee</u>. New York: Teachers College Press.

Watt, D. (1983, October). Mind-mirroring software. <u>Popular Computing</u>, pp. 65-66, 68.

Woods, W. A. (1970) Transition network grammars for natural language analysis. <u>Communications of the Association for Computing Machinery</u>, <u>13</u>.

Young, M. (1984). Information technology and the sociology of education: some preliminary thoughts. <u>British Journal of Sociology of Education</u>, <u>5</u>, 205-210.

Appendix

This appendix contains some articles from the Computer Chronicles News Network. All of the articles were written by students using computers and were sent via electronic mail through the Source (PARTI: CCNN), a commercial information utility.

(Lincoln Vista, California, October 22, 1984)

Article for section on Fashions

The clothing in Vista is probably very different than he kinds of clothing you wear in your country. In Vista the girls like to wear floresant colors. Personaly I don't think they are that exciting but I am not the one wearing them. Mini-skirts are also popular but I have noticed that they are slowly dieing out.

The guys wear Levis (501's) and they usualy roll the legs up so that they are known as high waters. Hightops are also very popular for guys. They are shoes which come above the ankle.

This concludes my article on Fashions. I hope you like it.

By Marcie Teuber

(Harbor View, Juneau, AK, 4-24-84) New Store Opens

They are putting up a Fred Meyer shopping center in Juneau. There are only two other shopping centers that can be driven to in Juneau. We either need a boat, or a plane to go enywere else. A lot of people are excited about this, becouse things like this hardly ever happen in Juneau.

By Pete Ellis, Grade 6

(Kamehameha Honolulu Hawaii, 13-Mar-84)

Sashimi

Sashimi is a Japanese type of food. Anybody can get it. It contains raw fish. The best kind of raw fish is AHi(Tuna). You could also make it out of Maguro(Sword Fish) or AKu (another type of Tuna). Sashimi is a red colored fish. It is made by cutting the raw fish into small and thin slices. You do not have to cook it. You eat it as an appetizer. In Hawaii we call it pupus. There is a sauce you eat it with. The sauce is made of hot mustard and shoyu(soy sauce). Most people like to eat it at New Year's Eve. That is the most expensive time to get it. You pay about $20.00 a pound, but people still buy it. Sashimi is my favorite appetizer. If you ever come to Hawaii and you go to a nice restaurant ask for Sashimi as an appetizer.

By Ana Vidinha, age 10

(Our Lady of Mercy College Parramatta, New South Wales Australia, Friday, October 19, 1984)

A Special Birthday

Today is our principal's birthday, whose name is Sister Janet. Yesterday we collected 20 cents from each pupil to buy her a present. We hope that she will let us out early today as her present to us.

She will be leaving us next year in August to study in the United States. It will be an exciting experience for her, and we will miss her very much.

By Gabrielle and Nicky