

RESEARCH IN WORD PROCESSING NEWSLETTER

Dr. Bradford A. Morgan and Dr. James M. Schwartz, Editors
South Dakota School of Mines and Technology Rapid City, SD 57701 Phone: (605) 394-2481

Vol. 3 No. 4

April 1985

Computers, Word Processing, and the Teaching of Writing

Brian Gallagher

The next few years will be the period in which educators will need to work on assessing the precise capabilities of microcomputers--including their all-important word processing functions--as pedagogical instruments. In each subject area, educators will have to create a detailed taxonomy of tasks which the microcomputer *might* help students to perform more efficiently, more accurately and better--and then analyze very carefully just which of these tasks the microcomputer does and does not lend itself to. In those cases where the computer allows for much faster work, educators should ask if it also encourages a more extensive grasp of the subject and whether the microcomputer provides ways of putting the time it saves to fruitful educational use. Educators will also need to calculate what *not* having access to a computer will mean in an educational system which will soon apparently be dominated, at least in its middle and upper reaches, by computer-related work.

One likely result of the computer revolution in education will be a reconceptualization of the teaching function. Many teachers will find themselves doing much more one-to-one teaching, leaving the rote instruction to the machines, which usually do this sort of thing better and can structure instruction to fit individual abilities, questions and needs.

Thus, there will be a coming together of teaching and tutoring functions, with both teachers and students having to learn the most effective points at which to consult about the students' ongoing work. However, teachers need not worry overmuch, under these changed circumstances, about being displaced by computers, for there will always be ample work for them to do. The teaching of writing using computers--as I have discovered through three years of teaching a variety of writing courses in which students wrote and revised all their work via word processing--is a good case in point.

As Peter Elbow describes it in his valuable, if slightly mistitled book, *Writing Without Teachers*, "Editing means figuring out what you really mean to say, getting it clear in your head, getting it unified, getting it into an organized structure, and then getting it into the best words and throwing away the rest" (Elbow, 38). This description of editing far exceeds the much more literal description of the "editing" function that can be found in the manual of any word processing program--and with good reason. It is precisely because writing teachers can intervene pointedly and effectively to help students learn editing in the complex way Elbow characterizes it that there will be a continued need for such teachers. Even if, as Helen Schwartz observes, "computerized text analysis can sometimes

Material for this essay is adapted from my book, *Microcomputer Word Processing Programs: An Analysis and Critique of Their Uses in Higher Education*, which has just been published by the Instructional Resource Center of the City University of New York.

2--Research in Word Processing Newsletter

see surprising and problematic configurations in writing that a human master can miss (even when the master knows what he or she is supposed to do)," it is nonetheless true that "writers [must] proceed after text analysis to the human hammering out of meaning" (H. Schwartz, 142). For the foreseeable future, writing will continue to be, ultimately if not always initially, a communication between one human agent and another.

Because it is not interactive, and not a direct teaching program, word processing is perhaps easier to assess as an educational aid than direct computer-assisted instruction. In a recent, important study of the writing process, the researchers concluded:

Good writers respond to *all* aspects of the rhetorical problem. As they compose they build a unique representation not only of their audience and assignment, but also of their goals involving the audience, their own *persona*, and the text. By contrast, the problem representations of the poor writers were concerned primarily with the features and conventions of a written text, such as the number of pages or magazine format.... Our expert writer simply...[spent] more time than the novice in thinking about and commenting on the rhetorical problem, as opposed to spending that time generating text. (Flower & Hayes, 29)

Because it automatically handles potentially distracting matters of format, layout, page length et al. and because it is simpler to generate neat, legible text using it, word processing can free student writers to concentrate on analyzing those aspects of the writing process that these researchers note "poor writers" ordinarily ignore in favor of more literal and less substantial ones. Mimi Schwartz puts the matter of how computers can help student writers generally and succinctly: "Given what current research tells us about composing, particularly revision, this machine seems both psychologically and technologically suited to help the writer write more and risk more and achieve more fully developed writing" (M. Schwartz, 27). This help, however, is not restricted to the revision stage -- or rather, it consists in orienting the writer to the act of revision throughout the writing process, thereby making the writing genuinely accretive, with the best of it preserved and polished and the rest of it discarded.

In many cases, however, it will in large measure be what else besides direct work on the computer is done in a writing course that will determine just how useful writing on a computer can be for students. For example, one of the most successful assignments developed by my colleagues and myself for our microcomputer/basic writing class consists of pairing students to interview and write essays about each other. After each stage, from completing a questionnaire (by hand) to finishing a penultimate draft, students exchange papers with their partners, allowing for the correction of factual errors, the filling in of omissions, and the refinement of phrasing and organization. The assignment helps create a mutual sense of audience--the novelty of, for once, having their "subject" speak back to them gives the student writers a good sense of the affective nature of their characterization of another person; and being a very interested reader fosters a strong sense of how oneself and others can and do react to written communication. In all this, the computer plays only, if importantly, a supporting role--quickly supplying multiple copies of a draft for simultaneous reading by both parties, enabling insertions and corrections to be permanently made as the need for them is discovered, and generally facilitating the evolution of a progressively more accurate and more complex characterization.

One writing teacher asserts, "I do not see how a composition course can fail to confront students with the question of *how* they know what they want to say, i.e., with epistemology, if only to the extent that it makes them aware that language is at once a tool and a trap" (Gage, 621-2). His point is well-taken, provided we add the qualification that such epistemological "confrontations" become the more complex as the writing task becomes the more difficult and far-ranging. For beginning writers word processing presents a very elemental, and very useful, confrontation: for perhaps the first time, students can face a legible, unarguable copy of what they have written and must learn to stand by exactly what they have asserted or revise it.

Using a word processing program as part of a formal course structure, rather than just making the equipment and software available to students to work on their own, allows for the immediate implementation of things learned in the classroom. For instance, revisions suggested by collaborative group tasks like peer-critiquing can be immediately (and hence more thoroughly) implemented when a writing session at a computer directly follows one in a classroom. The ability of any program to cut into a text at any point and to allow instant, permanent changes permits students to utilize more of what they have learned in the classroom than reworking their essay many hours (or a day to two) later--more even than if they immediately started revising the essay by hand, where changes are slower to make, harder to read, and more difficult to see and judge because they clutter the page. If, as one partisan of word processing puts it, "most first drafts, in fact, can be cut by fifty percent without losing anything

Research in Word Processing Newsletter--3

organic" (Zinsser, 99), then using a technology that entirely eliminates the cuts and neatly reformats remaining text is bound to be a better way of enacting whatever revision approaches and strategies the students are taught.

Other attributes of word processing could be used to implement lessons on, for example, the use of vivid, physical details in writing description. Students might have the program search their text for those vague words ("good", "bad", "happy" and, worst of all, "nice") that inexperienced writers tend to overuse--and they could then replace them with exact, evocative words and phrases. Similarly, students might learn some important things about organization by taking an essay--one of their own, which would be preserved in its original form in another copy, or one the instructor had put on their disks--and by moving the paragraphs around, printing out for analysis each of the different versions they had evolved.

Effective writing strategies, like free writing and pre-writing, can just as easily be performed on the computer as with pen and paper, perhaps even a bit more easily once students are comfortable on the machine, for the keyboard is light and quick to the touch, and it produces a more readable (if often typo-laden) copy at the end of the exercise. A few programs have preset formats that can be employed for creating outlines; and all programs are so flexible with insertions and movements of text that they lend themselves easily to the sometimes erratic task of making a complete and understandable outline. (Working on a computer spurred most of my students to make more thorough outlines than they ever had on paper.)

Stephen Marcus, of the South Coast Writing Project and the University of California at Santa Barbara, reports notable success with "invisible writing," a simple, clever technological variation on free writing:

Simply by adjusting the brightness knobs on their monitors, students can eliminate immediate visual feedback yet still record their ideas. The text may be eventually examined by brightening the screen display, and as usual, it is available for editing, saving and printing....students noted their usual pattern in composing was to interrupt the flow of thought frequently to edit and amend the language, syntax and mechanics of their text. Invisible writing experiments suggested to them that their usual pauses obstruct their fluency and, more importantly, dilute their concentration. (Marcus, 156)

Like Marcus', other effective pedagogical techniques employing the computer derive from a recognition of the machine's special capacities (and limitations). For instance, it is productive to have students keep a writing journal on the same disk as their writing assignments, so they can record thoughts about the act of writing and the particular writing they are doing, making entries just before or just after doing that writing--or even, when a crucial thought strikes them, in the middle of their writing.

As Robert Scholes observes, "Narration is, first of all, a kind of human behavior" (Scholes, 57). One of the best exercises to bring students to an awareness of this important fact--breaking them into small groups and having them compose a group narrative, with students contributing one sentence in turn until the narrative is completed--works particularly well using a computer, which gives more substance to the effort by displaying the results on a screen, by focusing the single writer's and several onlookers' attention on the display where the evolving text is being recorded, and by providing multiple copies for further group analysis and amendment. Those programs that allow "windows" to be opened, in which material from a second file can be put on the screen while text can still be entered to the first, are very useful for "writing by imitation" exercises: the text to be imitated can be put in one window and scrolled up as the student writes his or her imitation of it in the other window.

The simple capacities of most word processing programs, to say nothing of the sophisticated capabilities of prose analysis programs like *Writer's Workbench* and *Grammatik*, can be employed in ways that will help draw attention to any number of typical student writing problems. For instance, consider what Mina Shaughnessy discovered about the vocabulary of the basic writers she studied:

Three features of vocabulary stand out in papers of the basic group: a preponderance of vague nouns and ambiguous pronouns, a dependence on basic verbs, and an absence of modification. For reasons that are not always clear, or the same, writers at this basic level often fail to name the object or person or idea they are writing about. (Shaughnessy, 199)

With even a simple word processing program, a student could learn to use the "search" function to locate and replace instances of these first two kinds of vocabulary weaknesses--and even, by putting his or her topic into one

4--Research in Word Processing Newsletter

or two words (an instructive task in itself), have the program search to see if the topic is ever directly mentioned and, if so, how often. A prose analysis program would locate all these weaknesses, suggest suitable revision strategies and offer likely substitutions in many instances. Some prose analyzers, like *Lancelot* (developed at Clarkson University), even allow the writing instructor to decide which errors shall be flagged and to write the text of the error message, thereby permitting the program to be tailored to the specific problems and weaknesses of any group of writers. In the future, working with computers, writing instructors will need to spend less time marking things like "Frag," "Wrong Word" and even "Awk" on student papers and will have more time (and space) for substantive comments.

It may be taken as a general rule that the more advanced the writing course, the less extensive the benefits of using word processing in it, the exceptions being business writing courses and some technical writing courses, which naturally profit from word processing's primary development as an instrument of business. As writers become more sophisticated, they learn to put aside concerns with format and layout until the later part of the writing process and to see outlining, pre-writing and even thinking about their writing task as a means of generating more significant finished text and saving time in the long run. They also learn to take on larger, more complex kinds of writing tasks, and using word processing will in no direct way aid them in abstracting and generalizing, arguing for example, doing research and performing other procedures that must be learned as writing assignments become longer and more difficult. However, it *will* continue to provide them with a clean, instantly revisable copy of their evolving text--and so enable them to see just where they are so they might judge just where they want to go. No matter what the level of the writing course, students, properly guided by computer-knowledgeable instructors, will simply, but significantly, be able to expend more effort on each piece of writing--and spend less time doing so--than when writing by "older" means.

* * * * *

History shows us that new, vastly more efficient technology normally brings almost as many problems as benefits, the most significant of the problems in a profit-oriented marketplace being massive unemployment for workers in outmoded trades. Certainly the complacent belief that many of the workers thrown out of work by the rapid shrinking of American heavy industry over the last two decades would eventually find work in the booming electronic and computer industries is belied by studies which indicate that only 2% or 3% have yet been able to do so.

A point that is sometimes overlooked in technological history is that even persons who benefit from the new technology also face certain attendant problems. For all its considerable advantages, word processing is not without problems for the student user and, indeed, for any user. The most obvious disadvantage is lack of access. Economically disadvantaged students by and large have to rely on institutional computer resources which, no matter how ample, are always over-subscribed. Even if a student is rich enough to afford a personal computer, and even if that student is fortunate enough to possess equipment that is compatible with that at his or her institution, there will still be places--the library, study areas, locations off campus and away from home or dormitory--where the students needs to write and must do so by hand or typewriter. For students wholly conditioned to writing on a computer, writing any other way may prove frustrating and unproductive. "Machine dependency" is a real possibility, especially for weak writers, who sometimes see their noticeable improvement in writing as tied "magically" to their use of the computer to write. At my college, for instance, we are finding that some students are postponing taking the next required course in the writing sequence until they can take it in one of the special microcomputer/writing sections.

A study being undertaken at the University of Minnesota by Lillian Bridwell and Donald Ross has found that the characteristics of the word processing apparatus, like those of virtually all apparatuses, create problems for certain users:

...the limited size of the CRT screen (typically eighty characters across and twenty-four lines down) makes it difficult for writers to see as much of their text at one time as with paper and pen; as a result, text coherence may be hurt. Another problem is that the gadgetry available on a computer may distract some writers, causing them to break off from writing every few words to "play" with editing features. And, of course, no matter how simple the word processors are made, there will be a few writers terrified of the new medium. (Wresch, 798).

There is no question that writing on a computer is a more cumbersome activity in terms of the equipment needed to perform it than writing by any other means. For some writers, the sheer physical dimensions of the impedimenta

Research in Word Processing Newsletter--5

can be intimidating, especially if the writer is doing only a short, simple piece of writing, the small size of which seems out of all proportion to the equipment deployed in its creation.

While many students do make more substantial revisions when they write via word processing programs, there are, and probably always will be, a certain number of student writers who use the machine for what one computer-knowledgeable writing teacher terms "smokescreen revision--the tendency to think that nice appearances cover up flaws in meaning, and that facelifting changes are a substitute for changes in meaning" (M. Schwartz, 29). The neat, well-ordered appearance of the word-processed text can beguile some students into only looking for and correcting those mistakes--letter inversions, misspellings, unjustified margins, omitted words--which mar the appearance of the text. In the process all the major problems with coherence and larger meaning get ignored.

Even students who have learned to revise and correct wisely and well on the word processor may have difficulty using their newfound skills if there is not sufficient open time to use the institution's computers to allow them to complete their assignments. Good, effective revision, even with a computer, takes time. Once students have learned to use a word processing program, they need sufficient access to computers to insure that they will continue to be able to use this generally more effective and always more efficient way to write. Ideally a certain number of computers should be put in a room of their own, where they are always available for individual student use. Failing that, at least three or four computers in a lab/classroom should be set aside for individual student use (and, preferably, set off from the other machines, so as not to cause distractions for either the students working individually or those taking a class). Filling a computer lab with classes that use all of its terminals in every period of the class day may appear economical and cost-effective to administrators, but allowing no open time for practice of the computer skills being learned will seriously limit the ultimate value of the instruction being provided.

Another problem, and a more philosophical one to be sure, is that many of today's students take to writing on a computer almost too easily, too unreflectively. In my experience, "computer anxiety" (to say nothing of computer-phobia) is confined almost entirely to faculty. Students can often be blase about writing on a computer, totally unconcerned about the fact that they may be composing by radically new means and that old notions of "writing," "a text" and "to print" are being reshaped drastically in the process. Moreover, the unquestioning acceptance of computers as writing instruments not only creates some barriers when it comes to understanding "writing" in and of the past, but it also does not portend well for the future. There will always be some functions that these machines do less well than humans, and there will always be some people who would have these machines perform such functions anyway. Without the habit of questioning the computer's efficacy and suitability in each and every instance, society might become computerized to the point of absurdity, if not malignancy. Scrutinizing the educational uses of computers as writing instruments would be as good a way as any to initiate the process of critical review.

Works Cited

- Elbow, Peter. *Writing Without Teachers*. New York: Oxford Univ. Press, 1973.
- Flower, Linda and John R. Hayes. "The Cognition of Discovery: Defining a Rhetorical Problem." *College Composition and Communication*, 31:1 (Feb. 1980), pp. 21-32.
- Gage, John T. "Philosophies of Style and Their Implications for Composition." *College English*, 41:6 (Feb. 1980), pp. 615-22.
- Marcus, Stephen. "Real-Time Gadgets with Feedback: Special Effects in Computer-Assisted Instruction." *The Writing Instructor*, Summer 1983, pp. 156-63.
- Scholes, Robert. *Semiotics and Interpretation*. New Haven: Yale University Press, 1982.
- Schwartz, Helen J. "Monsters and Mentors: Computer Applications for Humanistic Education." *College English*, 44:2 (Feb. 1982), pp. 141-52.
- Schwartz, Mimi. "Computers and the Teaching of Writing." *Educational Technology*, Nov. 1982, pp. 27-29.
- Shaughnessy, Mina. *Errors and Expectations: A Guide for the Teacher of Basic Writing*. New York: Oxford Univ. Press, 1977.
- Wresch, William. "Computers and Composition Instruction: An Update." *College English*, 45:8 (Dec. 1983), pp. 794-99.
- Zinsser, William. *Writing with a Word Processor*. New York: Harper and Row, 1983.

[ED. NOTE: Brian Gallagher received his Ph.D. from the University of Pennsylvania and is now an Associate Professor of English at Laguardia Community College, City University of New York, 190 Sixth Avenue, Brooklyn, NY 11217.]

6--Research in Word Processing Newsletter

Bibliography Update

- Aitken, A.J., R.W. Bailey, and N. Hamilton-Smith, eds. *The Computer and Literary Studies*. Edinburgh: Edinburgh University Press, 1973.
- Bara, Bruno G. and Giovanni Guida, eds. *Computational Models of Natural Language Processing*. New York: North-Holland, 1984.
- Bassler, Richard A. "Word Processing for Educators and Researchers Using Very-Low-Cost Microcomputers." Western Educational Computing Conference: 1983. (California Educational Computing Consortium) North Hollywood, CA: Western Periodicals, 1983, pp. 147-153.
- Beil, Don. *The Bank Street Writer Book: The Rosetta Stone of Word Processing* (Apple Version). Reston, VA: Reston, 1985.
- Card, Stuart K. "Studies in the Psychology of Computer Text Editing Systems." Unpublished Dissertation. Mellon University, 1978.
- Chamberlain, William and Thomas Etter. *The Policeman's Beard Is Half Constructed: Computer Prose and Poetry by Racter*. New York: Warner, 1984.
- Coyle, William. "Using a Word Processor." *Research Papers*. Sixth Edition. Indianapolis: Bobbs-Merrill, 1985, p. 179.
- Degnan, Sarah C. "Word Processing for Special Education Students: Worth the Effort." *T.H.E. Journal*. 12:6 (February 1985), pp. 80-82.
- Embley, David W. and George Nagy. "Behavioral Aspects of Text Editors." *Computing Surveys*. 13:1 (March 1981), pp. 33-70. (Association for Computing Machinery).
- Heines, Jessie M. "Text Editors." *Screen Design Strategies for Computer-Assisted Instruction*. Bedford, MA: Digital Press, 1984, pp. 116-121.
- Holmes, Leigh Howard. "Follow 2: A Theme Commentary Program." *The Computer Teacher*. 12:5 (February 1985), pp. 38-39.
- Lanning, Gerald W. "Simplifying Documentation." *Computerworld*. 19:4 (January 28, 1985), pp. 37-38.
- Lannon, John M. "Writing for Computer Users." *Technical Writing, Third Edition*. Boston: Little, Brown and Company, 1985, pp. 590-593.
- Leonard, Rosemary. *The Interpretation of English Noun Sequences on the Computer*. New York: North-Holland, 1984.
- Levin, Robert and Claire Doyle. "The Microcomputer in the Writing/Reading/Study Lab." *T.H.E. Journal*. 10:4 (February 1983), pp. 77-79, 100.
- Marius, Richard and Harvey S. Wiener. "Writing with Word Processors." *The McGraw-Hill College Handbook*. New York: McGraw-Hill, 1985, pp. 572-574.
- Modern Language Association. "New Software for Bibliographic Search." *MLA Newsletter*. 17:1 (Spring 1985), p. 5.
- O'Malley, Christopher. "Writing Smarter: Knowing the Shortcuts and Print Enhancements in Word Processing Can Save Countless Strokes While You Polish the Printed Page." *Personal Computing*. 9:3 (March 1985), pp. 51-54.
- Pickett, Nell Ann and Ann A. Laster. "Computer Graphics." *Technical English: Writing, Reading, and Speaking*. Fourth Edition. New York: Harper and Row, 1984, pp. 496-497.
- Rethje, Linda. "Moving Writer and Reader into a New Literary Medium." *The Computing Teacher*. 12:5 (February 1985), pp. 61-68. (Reviews of elementary-school creative writing software)
- Shore, Howard A. "Bell Labs Develops a Program To Help Writers Improve Style." *Softalk*. 4:6 (February 1984), p. 230.
- Sullivan, David R. et al. "Advanced Word Processing." *Computing Today: Microcomputer Concepts and Applications*. Boston: Houghton Mifflin, 1985, pp. 135-155.
- Tritten, Larry. "I Do Not Process Words! How a Tradition-Conscious Writer Copes with Computers." *Popular Computing*. 2:7 (May 1983), pp. 118, 122.
- Wisbey, R.A., ed. *The Computer in Literary and Linguistic Research*. Cambridge: Cambridge University Press, 1971.
- "Word Processing in the Classroom." *ATARI Teachers' Network*. 1:2 (July 1983), p. 2.
- Worley, Lloyd. "Using Word Processing in Composition Classes." Annual Meeting of the Conference on College Composition and Communication. New York, NY: March 29-31, 1984. [ED 243 127]. 10 pp.

--B.A.M.

Research in Word Processing Newsletter--7

The Scholar's Software Library

Bryan Pfaffenberger

Program:	<i>Framework</i>
Available From:	Ashton-Tate 10150 W. Jefferson Blvd. Culver City, CA 90230 Phone: 213-204-5570
Price:	\$695.00
Requires:	IBM PC, PC XT, or 100% PC-compatible; minimum 256K (384K recommended; 512K or more required for lengthy documents)

Unless you've spent the past eight months collecting shells on the beaches of Sri Lanka, you've doubtless heard of the epic showdown between Ashton-Tate's *Framework* and Lotus Development Corporation's *Symphony*, the two big--and expensive--integrated programs for the IBM PC. Both offer all five software functions (spreadsheet, word processing, database management, graphics, and telecommunications) in single, memory-hungry packages.

Those whose primary interest lies in word processing have watched this showdown with, at best, mild interest. After all, cramming all those software functions into a single program necessarily means that features have to be trimmed. *Symphony's* word-processing module has been pared down so far, in fact, that it is little more than a rather cumbersome adjunct to the program's ultra-sophisticated spreadsheet. Most writers have doubtless concluded that they would be best off with a full featured word-processing program rather than an integrated package.

But *Framework's* word-processing module is another matter altogether. Even those writers who may never use *Framework's* spreadsheet, database management, telecommunications, or graphics modules should have a look at the program. It is, simply put, a magnificent tool for writing. There is no commercially available program that I know of which gives a writer as much control over the structure and organization of a document.

Framework, to be sure, does not offer all the features one would expect in a sophisticated, state-of-the-art word processing program such as *Microsoft Word* or *WordStar 2000*. It does not include, for instance, footnoting, online spelling checking, or glossaries. Experienced word processors will find annoying the lack of certain formatting features, such as superscripted or subscripted numbers. And if you're interested in *Framework*, start looking for a cheap source of 64K RAM chips: you'll need 512K to cope with a 73-page document.

Yet *Framework's* strengths are sufficiently impressive that, for most users, its shortcomings pale in significance. Even by itself, the word-processing module is impressive. It reminds one forcefully of *Microsoft Word*, sans snazzy features and mouse. A Macintosh-like user interface, complete with pull-down menus, makes the program remarkably easy to use. And because the entire document is present in the main memory, it's fast--very fast. But it is the program's overall conception that, for writers, will prove most appealing of all.

At the heart of *Framework's* design is an idea processor that goes far beyond its predecessors, such as *ThinkTank*. A typical *Framework* writing session begins with the construction of a *ThinkTank*-like outline:

8--Research in Word Processing Newsletter

1. Introduction
2. The Problem
 - 2.1 Statement of the Problem
 - 2.2 Survey of Relevant Literature
 - 2.2.1 Literature before 1970
 - 2.2.2 Literature 1970-1980
 - 2.2.3 Recent Literature
- [...]
5. Research Results
 - 5.1 Summary of Results
 - 5.2 Analysis of Results
 - 5.3 Graphic Presentation of Analysis
6. Conclusions
7. Appendix: Research Results

Figure 1: A Framework Outline

As each outline heading is created, *Framework* automatically establishes a frame which is attached to it. A frame is nothing more than an expandable box into which information can be inserted. To write the section on "Literature Before 1970," for instance, you would press a function key which opens the frame and expands it to fill the screen. To work on the next section, you would give additional commands which close the frame you've just finished writing and open the next frame. A *Framework* document, then, is made up of a linked series of frames.

What makes this approach so fruitful is the ease with which you can switch back and forth between the outline and the text. In *Framework's* outline mode, your document is shown as an outline. In its text mode, however, you see a full-screen display of the text, which looks just like a document prepared with any word processing program save that the frame boundaries are visible at the beginning and end of the frame. A single keystroke permits you to jump from the writing you're doing to an outline of your document, helping you to keep its overall organization clearly in mind, and another keystroke takes you back to the text. When you're finished writing, the entire document may be printed as if the frame boundaries did not exist: neither the frame boundaries nor the headings appear on the printout.

Writing a document this way--putting each section and subsection in its own frame--gives you an amazing amount of editing power. Instead of editing the document by moving blocks of text from one part to another, you can simply edit the outline, with dramatic effects. With only a few keystrokes, it is possible to achieve a major reorganization of a document, and all the while the overall plan and structure is plainly visible. The outline is automatically renumbered as its components are rearranged. For this reason, working with *Framework* is an entirely different experience from working with a preconceived outline. Invariably, writers find reason to depart from outlines created at the beginning of a writing session, so the outline becomes an increasingly inaccurate guide to the document's structure. With *Framework*, the outline is always accurate.

Framework handily overcomes what I believe to be the major shortcoming of all other word processing programs: the barriers they impose on getting a clear view of the overall shape and organization of a document. Whatever the flaws of writing with a pen or a typewriter, at least you can see where you've gone and where you're headed just by shuffling through the pages. With a word-processing program, however, you're looking at only one-third of a screen at a time. Many programs scroll at a pace so glacial that there is precious little incentive to look back through what you've written. And the result, all too often, is a document whose organizational flaws are not visible until it has been printed.

Research in Word Processing Newsletter--9

Framework has truly realized the potential inherent in the idea-processing concept. But there's more. *Framework* (in contrast to *Symphony*) permits true integration of documents: frames can contain not only text, but also spreadsheets (which print out as neatly formatted tables), graphics, and databases. The document outlined in Figure 1, for instance, could incorporate all four ways of representing information. And what is more, all four can be printed at the same time:

[...]

5. Research Results [TEXT]
 - 5.1 Summary of Results [TEXT]
 - 5.2 Analysis of Results [SPREADSHEET TABLES]
 - 5.3 Graphic Presentation of Analysis [LINE CHART]
6. Conclusions [TEXT]
7. Appendix: Research Results [DATABASE PRINTOUT]

Figure 2: Combining Text, Spreadsheet, Graphics, and Database Frames in a Single Document

Quantitative data, moreover, can be tagged so that changes to the data in one section of the document will be automatically reflected everywhere the data appears, and any calculations or regraphing operations which are consequent upon the change will also be done automatically.

Framework is receiving praise for successfully integrating the chief software functions in a single, easy-to-use program. In my view, however, *Framework's* true achievement lies in the power of its tools for conceiving, writing, reorganizing, and printing complex documents. *Framework*, more than any of the "state-of-the-art" word-processing programs now available, truly points the way towards the word-processing software of the future.

[Ed. Note: Bryan Pfaffenberger--an anthropologist, college teacher, and writer--is the author of *The College Student's Personal Computer Handbook* and *Macintosh for College Students* (Sybex Computer Books, 2344 Sixth Street, Berkeley, CA 94710) as well as *The Electronic Scholar*, to be published this year by Little, Brown and Company. He is currently on leave from his position as Associate Professor of Anthropology at Knox College.]

Errors Caused by the Use of Word Processing in Reports by Beginning Technical Writing Students

John A. Stibravy and Chuck Beck

At the Air Force Institute of Technology, many of the engineering students are buying word processing systems at the same time they are enrolled in our Technical Writing course. Thus, it is possible to study the impact of word processing on errors in student papers. Specifically, we identified writing errors due to word processing from 150 student papers.

However, before we list the errors due to word processing, Table 1 identifies the 20 most frequent errors, regardless of cause, in the 150 papers.

TABLE 1

MOST FREQUENT ERRORS IN SAMPLE PAPERS

- Words were misspelled for the context.
- Conclusions were missing when needed.
- Figures were missing when needed.
- Punctuation was wrong.
- Subject-Verb agreement was lacking.
- Titles of figures were missing.
- Bibliography format was wrong.
- Page numbers were missing.
- Figures were numbered out of sequence.
- Figures were unprofessional.
- Style was improper for the audience.
- Margins were too small.
- Page numbers were missing in Table of Contents.
- Table of Contents was in wrong place.
- Conclusion was located before the final text figure.
- Format was not all the same.
- Figure Titles in Table of Contents were not the same as those in the actual figure.
- Sources were placed in figures in unlikely places.
- First letter of each word in figure title was not capitalized.
- Words were capitalized which should not have been capitalized.

Table 2 shows those errors due to the students' use of word processors.

TABLE 2

ERRORS DUE TO WORD PROCESSOR USE

- Words were misspelled for the context.
- Page numbers were missing in Table of Contents.
- Table of Contents was in wrong place.
- Conclusion was located before final text figure.
- Format was not all the same.

Conclusions

Although errors attributable to word processors comprised $\frac{1}{4}$ of the 20 most frequent errors, word processing errors were only 8% of the 12 most frequent mistakes.

The majority of mistakes made by students were not related to the use of word processors at all. The errors were neither caused by word processors nor could the use of a word processor solve the error, such as in the case of subject-verb disagreement or the bibliography format being wrong.

Summary

Teachers may find it useful to sample their students' papers and to derive a list of errors. It may also be useful to determine how many of these errors are attributable to the use of word processors and how many would be solved by switching from a typewriter to a word processor. Our results suggest that in the field of technical writing, merely switching from a typewriter to a word processor may not solve many student errors. In our sample, only error #10 (Figures were unprofessional) would clearly be improved if the writer had a graphics capability with the word processor.

However, teachers who use an error identification checklist should remind students that true quality in a document is not achieved by correcting surface errors, but through the proper implementation of quality *thought*.

Dr. John Stibravy is an Assistant Professor of Technical Communication who teaches Technical Writing at the Air Force Institute of Technology. Dr. Chuck Beck is an Associate Professor of Research and Communication who teaches Research Methods at the Air Force Institute of Technology. They may be contacted at AFIT/LSH, Wright-Patterson AFB, Ohio 45433-6583. (513/255-2254)

ZyINDEX -- State-Of-The-Art Text Management

In the history of microcomputer-software development, information-management landmarks have been reached. Unfortunately, these seminal discoveries (*VisiCalc*, *Lotus 1-2-3*, *DBase II*) have been directed to business-oriented users. If academic audiences wanted to exercise the power of these database managers and spreadsheets, they were forced to exorcise their habitual ways of doing things and format information such as bibliographies by adhering to the often dictatorial field- and record-size limitations imposed by the business-programming gurus. But no more.

One product that seems destined for greatness among teachers and researchers involved with information handling is *ZyINDEX*, a full-text indexing program published by ZyLAB Corporation (233 East Erie Street, Chicago, IL 60611; phone 312/642-2201). *ZyINDEX*, requiring 192k RAM, works directly with most popular PC-DOS and MS-DOS word processors by using "map files" which handle I/O operations between the respective word processors and the *ZyINDEX* program. The WP software currently "mapped" by *ZyINDEX* include the following:

EasyWriter II, *Microsoft Word*, *Multimate*, *Palantir*, *Smart Word Processor*, *Volkswriter Deluxe II*, *Wang Word Processor*, *WordPerfect*, *WordPlus PC*, *WordStar 3.x* and *WordStar 2000*, *Xywrite II*, and all pure ASCII-based programs such as *Final Word*, *PeachText*, *PerfectWriter*, *pfs:write*, *Select*, *SkiWriter II*, *SuperWriter*, *Textra*, etc.

As you can see from the above list, just about every major word-processing program is currently supported by *ZyINDEX*. And if yours isn't among these, a call to ZyLAB will be all that's needed to find out if it is one of the programs to be supported in the often-revised, ZyLAB map-file catalog.

Now to the interesting stuff. Just how easy is it to make *ZyINDEX* do what it does so well--give you a referencing and outputting handle on just about every word in every file you have? In a word, *very!* After following the simple program-installation instruction, you type **ZYINDEX**, and the magic begins. On the screen is the first of several clearly worded menus designed to facilitate your indexing files:

F10: Exit	F9: Help
INDEX MENU (1)	
Specify Drive and/or Pathname of Sub-Directory [A:]?	
Enter A: B: or C: etc. for the disk drive holding the files you want Indexed. A: is the default drive, so if your files are on drive a:, just press Enter. If your files are in a sub-directory, as may be the case on a Hard Disk, enter the complete pathname, ending in a "\\".	
For example: C:\jundir\accounting\	

12--Research in Word Processing Newsletter

Index Menu (2) is optional, allowing you to name a floppy disk for record-keeping purposes. Because *ZyINDEX* marks indexed files on the system level (thus, not altering your original text), it also keeps a record of all files which have been sent through the program for indexing.

The hard-disk directory we used to evaluate *ZyINDEX* is named "c:\t\," and typing that in resulted in the following screen full of information:

F10: Exit	F9: Help	F2: Index files	F3: Mark new files	PgDn: Skip files
Select Files to be Indexed: move cursor to file names and press <enter>				
CH09-RTI.LEC	CH19-RTI.LEC	CH01-RTI.LEC	CH02-RTI.LEC	
CH03-RTI.LEC	CH05-RTI.LEC	CH06-RTI.LEC	CH07-RTI.LEC	
CH14-RTI.LEC	MECH02.TST	MECH03.TST	MECH04.TST	
No files have been previously indexed with this path name & disk name.				

After choosing all of the files with the extension "LEC" (in this case, my technical-writing lecture notes), *ZyINDEX* highlights the choices and, when function key F2 is pressed, begins the indexing process. To keep you informed of its progress, *ZyINDEX* brings up another screen:

Esc: stop indexing after current file			
12% of 316 Words Completed Processing			Indexing. . .
CH09-RTI.LEX	CH19-RTI.LEC	CH01-RTI.LEC	CH02-RTI.LEC
CH03-RTI.LEC	CH05-RTI.LEC	CH06-RTI.LEC	CH07-RTI.LEC
CH14-RTI.LEC	MECH02.TST	MECH03.TST	MECH04.TST

As each highlighted file is indexed, you receive a running tally of the percentage of indexing completed as well as the total number of *content* words in your file. *ZyINDEX* makes an important distinction between words of substance (or content) and words which the program never indexes (called "noise words," these include "and," "or," "of," "he," "she," etc.). All three versions of *ZyINDEX* (I reviewed the professional version, which lists at \$295) allow you a 64,000-word index maximum per file and a 125,000-word maximum of unique words allowed per index list (the Plus version, which costs \$695 and includes a license for use on a network of up to five workstations, increases that limit to 500,000 unique words). Since "noise words" make up a large number of our everyday written documents, your actual file-size limit probably will exceed 100,000 words (or close to three completely full 360k floppy disks). And *ZyINDEX* can handle 500 (Standard), 5,000 (Professional), or 15,000 (Plus) *individual* files in which all content words are indexed ("Full-Text Indexing"). If you choose "Word-Only Indexing," which keeps track of words in all files but does not keep track of where the words are in the files, your limits are 1,500 (Standard) and 15,000 (Professional and Plus).

After your files are indexed, you return to the operating system and enter **ZYSEARCH** at the system prompt. A new program screen appears, ready to lead you through your search for information:

F10: Exit	F9: Help
SEARCH MENU	
F1	Input new search request
F2	Search
F3	Modify search request
F4	Review search results
F5	Display disk directory
PRESS THE APPROPRIATE FUNCTION KEY	

Research in Word Processing Newsletter--13

Pressing F1 sends you to the "Search Request Screen," on which you begin to discover the true power of the ZyINDEX system:

F10: Exit
SEARCH REQUEST SCREEN
To edit the Search Words or Phrases use the following keys: BACKSPACE key to ERASE a character or a Connector that was just typed in DEL key to DELETE the character or Connector above the cursor ARROW keys to move the cursor
Use the following keys to type Connectors:
Ctrl + a: AND Ctrl + o: OR
Ctrl + w: W/ Ctrl + n: AND NOT
Parentheses may be used to group items together. When the NOT Connector is used, it must be followed by a parentheses, e.g. car AND NOT (parking lot).
* is the Multi-Character Wild Card Symbol ? is the Single-Character Wild Card Symbol
communications

"Communications", the word highlighted above, was my search request. Of course, things can get a great deal more involved, search-wise, depending on the logical operators ("AND", "OR", etc.) you want to use in the search. Another interesting feature of ZyINDEX is its ability to search for word/phrase combinations that are within a certain physical proximity: keying in, say, "tree w/10 maple" will find every text string in which the words "tree" and "maple" are within ten words of each other. The program's upper limit for "n," in the statement "w/n," is a whopping 30,000!

Back to more simple matters. After telling ZyINDEX that "communications" was the search word, the "Search Menu" returns. Pressing F2, the "Search" key, quickly sends the software into warp drive, and under the requested search word is displayed how many files have been retrieved (how many of the indexed files contain the chosen search word). Your eyes behold the next ZyINDEX screen:

F10: Exit	F9: Help	Arrow keys: move cursor	PgDn: page down	PgUp: page up
Select the File to be Displayed: move cursor to filename and press <enter>				
CH09-RTI.LEC	CH19-RTI.LEC	CH02-RTI.LEC	CH03-RTI.LEC	
communications				
4 Files Retrieved				

The file I chose to view was CH19-RTI.LEC, and after pressing F10 I saw, excuse the expression, a carbon copy of my original WordPerfect textfile with the word "communications" highlighted:

14--Research in Word Processing Newsletter

F10: Exit	F8: Unmark	F7: Mark	F6: Save	F5: Find
Displaying File: c:\t\CH19-RTI.LEC				
<p>Preparation is important, but DELIVERY is everything. Many a speechmaker has taken a dull subject, or a series of half-truths, and made the audience wave flags, jump up and down, or otherwise believe what the speaker had to say was important. In technical communications, where emotional responses to presentations are not the norm, you will have to rely on your delivery technique to make your audience feel satisfied with your performance. Notice that I said "performance." Whatever your topic, whatever the professional context, you will be judged as much on how you present your material as on the actual material itself. Speeches are, by nature, full of generalizations and communications 4 Files Retrieved</p>				

At this point, *ZyINDEX* highlights your search word and allows you to "Find" (F5) all other occurrences of the word in the file as well as MARK (F7) any number of lines in your file for processing.

After choosing the text you want, a simple tapping of the F6 key prompts you to enter the filename and extension of the marked text, as well as the directory or drive on which it is to be stored.

Finally, you are sent to the "Review Search Results Menu," which offers you three options: display retrieved files (F1), select files for printing (F2), and begin printing (f3). At this point, *ZyINDEX* can handle the printing, or you can return to your word-processing program and let it do the work. The great thing about *ZyINDEX* is that the file you just synthesized from your original indexed files is in the format of your chosen "mapped" word processor. None of that silly straight ASCII importing and exporting stuff which takes the charm out of, say, enhanced, right-justified text.

That's all there is to it. As you can see, *ZyINDEX* rates an A+ in the ease-of-learning, ease-of-use categories. But even if it were difficult to master, the results would be well worth the effort: complete content-word indexing and retrieval of all the information in your files and the ability to use your own word-processing software to create documents in any format and of virtually unlimited indexing length (of course, most WP programs set length restrictions of their own which fall far short of *ZyINDEX*'s limits).

--JMS

Projects Underway for 1985 MLA Convention

Preparations are now underway to organize computer-oriented approaches to writing and literature at the Modern Language Association's 1985 convention in Chicago next December. Divisions, special sessions, and allied organizations making plans include the following:

- ▶ "Computers and Other Electronic Media: New Directions for the Language Lab." Contact Prof. Renate A. Schulz, Department of German, University of Arizona, Tucson, AZ 85721. (Teaching of Language Division)
- ▶ "User-Friendly Editions and Critics' Responsibility to Versions." (Critical response to versions through user-friendly scholarly editions) Contact Prof. Miriam J. Shillingsburg, Department of English, Mississippi State University, Mississippi State, MS 39762. (Methods of Literary Research Division)
- ▶ "Character-Set Standards for the Microcomputer in Academic Humanities Applications." Contact Prof. Daniel T. Brink, Department of English, Arizona State University, Tempe, AZ 85287. (Special Session)
- ▶ "Computers and the Concept of Text" (The impact of computing on thinking, reading, writing, and the teaching of writing from any philosophical, critical, or pedagogical viewpoint) Contact Prof. Jeffrey L. Spear, Department of English, New York University, New York, NY 10003. (Special Session)
- ▶ "Computers and the Technical Writing Classroom: New Implications for Teachers and Students." Contact Prof. Mary F. Lay, Chairperson, Department of Technical Communications, Clarkson University, Potsdam, NY 13676. (Association of Teachers of Technical Writing)

Research in Word Processing Newsletter--15

Association for Business Communications

The West-Northwest Regional Meeting of ABCA will be held May 2-4, 1985, in Scottsdale, Arizona. Papers on word-processing applications to business writing include:

- ▶ "Artificial Intelligence and Editing" (Mohan Limaye, University of Texas at Austin)
- ▶ "Researching the Effectiveness of Computer Text Analysis for Improving Business Writing" (Mildred I. Johnson, Colorado State University at Fort Collins)
- ▶ "Quick and Easy Word Processing for the Busy [sic] Communication Class" (John J. Brugaletta, California State University at Fullerton)

Contact Jenny Gilsdorf, Convention Chair, Department of General Business, College of Business, Arizona State University, Tempe, AZ 85287.

World Conference on Computers in Education

WCCE/85 will be held in Norfolk, Virginia, from July 29 to August 2, 1985. In addition to other wide-ranging presentations covering computers in education, some panel sessions will cover word-processing applications to writing and literature, including

- ▶ "Word Processing in Pre-College Education." (Cathy Conlin, Nancy Olsen, and M. Goldfarb)
- ▶ "Computers and Writing: Variations on a Theme." (Stephen Marcus, University of California-Santa Barbara, and Valerie Arms, Drexel University)
- ▶ "Using Computers To Teach Literature." (Nancy M. Ide, Vassar College; Frank Bochert, Duke University; Bates L. Hoffer, Trinity University; Lance Miller, IBM)
- ▶ "What Has CAI Done for the Humanities." (Joseph Raben, Paradigm Press; Glyn Holmes, University of Western Ontario; Donald Ross, Jr., University of Minnesota; Michael Arenson, University of Delaware)

Contact WCEE/85, c/o AFIPS, 1899 Preston White Drive, Reston, VA 22091

Manuscript Submissions Welcome

The newsletter welcomes article submissions which pertain to word-processing, text-analysis, and research applications in academic situations. Manuscripts may be submitted either as hard copy or on disk using *WordPerfect*, *WordStar*, or standard ASCII code in IBM PC-DOS, MS-DOS (5¼" diskettes), or CPT 8500 (8" disk) formats. The Editors reserve the right to edit articles, if necessary. If you want your manuscript or disk returned, please send enough postage to cover the return along with a self-addressed envelope. Address all correspondence to the Editors, *Research in Word Processing Newsletter*, South Dakota School of Mines and Technology, 500 E. St. Joseph, Rapid City, SD 57701-3995.

© 1985 South Dakota School of Mines and Technology

Research in Word Processing Newsletter. Volume 3, Number 4. Copyright © 1985 by the South Dakota School of Mines and Technology. All rights reserved. ISSN: 0748-5484. Also indexed by *ERIC* and *INSPEC*. The *Research in Word Processing Newsletter* is published 9 times a year (September through May) by the South Dakota School of Mines and Technology, Rapid City, South Dakota 57701-3995; telephone (605) 394-2481. Postage paid at Rapid City, South Dakota. **SUBSCRIPTIONS:** \$12 per year (US); \$18 per year (Canada); \$24 per year (foreign). Address all subscription inquiries and manuscripts to the Editors, *Research in Word Processing Newsletter*, South Dakota School of Mines and Technology, 500 E. St. Joseph, Rapid City, SD 57701-3995. Please allow four to six weeks for new-subscription processing.