Human-Computer Interaction Perspectives on Word-Processing Issues

Patricia Sullivan

Word processing has been vigorously studied by those interested in the field of human-computer interaction. Because it naturally brings a wide spectrum of new users to the computer and asks those users to adapt their work habits to the machine, word processing has been a testing ground for ideas about learning, about training, about documentation, and about interface design. Seven journals and five conferences devote extensive space to the study of human factors in computing, and word-processing studies are prominent. Yet, this literature is not widely used to inform the research into computers and composition.

True, those researchers interested in human-computer interaction generally hold goals different from those of us who study the impact of computers on composition. The human-computer interaction researchers want to answer questions about how difficult word processing is to learn, about the impact of user interface designs on new users, about how educational materials should be constructed and packaged for users, and about learning to compute. These goals are obviously tangential to ours, as those researchers do not carefully examine, or even much care about, the quality of the text that users create. Human-computer interaction researchers measure learning by improvement in the users' knowledge of program features, by reduction of errors, and by increased speed of use rather than by improvement in the quality of the writing. But these researchers should not be dismissed out-of-hand, as they have been carefully observing users learning to use word-processing software. Perhaps knowledge of their findings can provide alternative explanations for inconclusive findings in composition research or can add converging evidence for other findings.

This article introduces human-computer interaction in its own context first, then argues that understanding word-processing research done in that setting can enrich our own thinking about the impact of teaching writing with the use of computers. The article's plan is to sketch the goals of this interdisciplinary field, then to review some of its word-processing studies [2], and finally to suggest issues developed in these studies that may interest researchers in computers and composition.

The Contexts of Human-Computer Interaction

"Human-computer interaction" is a rather slippery field. Evidence of its polyglot nature can be found in the fact that not everyone involved calls it "human-computer interaction"; some call it "man-machine studies," others "computer-human interaction," and still others "human factors in computing."

The common ground of this group, however it is labeled, is the building of computer systems that are more responsive to human needs. It draws on interdisciplinary yokings of various fields (industrial engineering, computer science, cognitive psychology, linguistics, human factors, information science, cognitive science, and at some times document design and education) in the service of studying the interaction of people and machines. Early on, when the group was closer to ergonomics and human factors, studies focused on physical factors (e.g., the size of letters on a display screen). Current work is more attuned to mental action, though, as it takes up issues such as memorability, modeling of learning, and usability. The prominence of cognitive psychology does not mean that the field is evolving into a sort of applied cognitive science (i.e., a group that models how the mind learns computing tasks). Although some people pursue psychology-linked goals, the entire group does not privilege them.

A central split in attitude is a theory-practice split that can be clarified if we contrast the approaches of the cognitive psychologists and the engineers. The psychologists naturally aim to build theories of users or of learning, while the engineers usually aim to build systems that solve problems they notice. To the extent that human-computer interaction embraces its research problems, its psychologists become engineers and its engineers become psychologists. But in real life, the groups are uneasy partners.

A recent book, Interfacing Thought (Carroll, 1987), and its reviews illustrate the tensions. The book had psychologists present practical theory and then had discussion-critiques given by Reisner and by Whiteside and Wixon. Reisner (1987) argued that human-computer interaction is more engineering than applied science because building better computer systems for people is the focal problem. But she sounded somewhat uncomfortable making the argument. Whiteside and Wixon (1987) moved further from psychological models as they urged that the study be approached in the most realistic contexts possible. They cited an installation study that worked smoothly in a laboratory but was inhibited in real life by the boxes and packing that filled the customer's office. The lab and the well-managed studies, they claimed, made people blind to some real problems. These psychologists were working hard to be more practical.

But when Interfacing Thought was reviewed by Gray and Atwood (1988) in theSIGCHI Bulletin (journal of the Association for Computing Machinery's Special Interest Group on Computer and Human Interaction), it was charged with being too theoretical and with being aimed at psychologists:

[1]If the authors wish to bridge the gap between theory and application, the best approach is to demonstrate application of the
Gray and Atwood's review went on to say that they are not sure the book should be read by anyone other than cognitive scientists, because design engineers would not know how to deal with the articles. The only article they found accessible to designers was Whiteside and Wixon's.

The book and its review, then, expose some tensions present in the field. Although these tensions are not as evident in the word-processing research as they are in other aspects of the field, I have discussed them as a way of providing a context. Some arguments in the field may not make sense to the new reader of human-computer interaction studies unless the reader is aware of the subtexts. Still, while the subgroups may privately value theory over practice, or building over testing, or innovation over status quo, publicly they value "IT WORKS FOR PEOPLE." Their inextricable linking of system and people makes human-computer interaction valuable for those of us in composition studies.

The Word-Processing Studies Reflect Their Field's Issues

To understand another's gait, we must walk in that person's shoes. For that reason, this discussion proceeds cautiously, summarizing studies representative of major goals set for word-processing studies, then listing other relevant studies in Table 1 at the end of this article. This discussion purposely casts the talk in terms used by researchers in human-computer interaction rather than in our own terms, trying to help us "hear" their voices so that we may better read their literature.

As we might expect, given the context discussed above, the terminologies used by human-computer interaction researchers are not totally consistent, and the subgroups do not globally agree on research agendas. The subgroups do generally agree on why they study word-processing and editing programs, though. Those applications are studied in order to find out more about how people learn to use, adapt to, and deal with computers.

Word-processing studies done in the 1980s show that the field has been driven by its products. Early studies defined the key features and developed generic ways to evaluate editing and word-processing programs and their interfaces. Then, studies began to focus on the task of training (including technological developments). As technology began to develop new interfaces, the studies began to shift back to evaluating the new features, with important new interests in group cooperation and graphics. In a sense, though, the human-computer interaction researchers interested in word processing encourage the technological advances as well as react to them. Their research into developing features (e.g., the use of menus) encouraged designers to modify designs in certain ways. In a curious way, designers drive the features while the features drive them.

Word-processing studies [3] can usefully be classified by the goals they pursue:

- **Goal of Improving Training/Education** To teach people to use editing programs, word-processing programs, or special features, and to evaluate materials or programs designed to train people.
- **Goal of Understanding Learning (Skill Acquisition and Modeling)** To build an understanding (or a model) of how people learn to use editing or word-processing programs.
- **Goal of Improving User Interface Design** To study features of interfaces (or the part of the program exposed to the user) in order to develop better interfaces.
- **Goal of Evaluating and Developing New Products** To study features of emerging technology in order to discover how easily people will use them, and to evaluate current products for human factors (e.g., ease of learning, ease of use).

These goals, of course, are not mutually exclusive. A study may, for instance, have people use several types of training materials. If the goal of the study is to improve training materials, it is classified a training study; if the goal is to make use of several ways to learn, then it is classified a learning (or skill acquisition) study. But the study may still speak to both goals.

**Goal of Improving Training/Education**

Training studies have had a place in the human-computer interaction literature primarily because word processing is connected to the office. Because of that link to office automation, training is the general approach taken to the teaching of word processing. Training studies in human-computer interaction tend to focus on developing on-line help, computer-based training, manuals, or interface changes as remedies to training problems. One of the most discussed approaches to training is the minimalist training approach developed at IBM Watson (Carroll, 1984).

The minimalist approach downplays passive instructions and urges active learning. A study of one minimalist manual (Carroll, Mack, Lewis, Grischkowsky, & Robertson, 1985) reports the success of that manual. The manual urged exploration through a number of features: less reading, greater task orientation, more learner initiative, more error recovery information, and easier reference. Word-processing users of the minimalist manual were 40% faster at covering the basic topics, were as good on the achievement tests, and were better in tests for self-sufficiency than users in other training conditions. In a study of the minimalist interface, Carroll and Carrithers (1984) suggest supplying "training wheels" for the interface to make the learning easier. By blocking inappropriate, complex, and wrong choices, the training wheels interface limited the number of possible problems encountered. It forced new learners to learn the simple actions first (e.g., typing, editing, and printing) while blocking advanced functions (e.g., data merging, pagination, spell checking, and format changing). The interface study asked 24 learners to use either the training wheels or the commercial system to type and print a simple document. The training-wheels users got started faster, produced better work, spent less time on errors, and understood concepts better. Thus, the Watson researchers found the minimalist approach useful in both helping the active user and in blocking
Goal of Understanding Learning

In this decade, there has been continual study of how people learn to use word-processing programs. The psychologists involved in human-computer interaction have focused on learning (or skill acquisition) more than on the other goals, perhaps because skill acquisition ties them to traditional psychology. A new and “hot” topic is skill transfer. Since 1985, many researchers have been studying how difficult it is for people to move to a new word-processing program. But the older themes of how difficult it is for people to learn word-processing programs, themes which originally surprised the human-computer interaction group, are robust, as well.

Mack, Lewis, and Carroll (1983) enumerated many learning difficulties when they used protocols to study a number of problems and issues that now reverberate through the literature: 1) learning is difficult; 2) learners don’t know how computers work; 3) learners make up interpretations for what happens; 4) learners generalize from what they already know; 5) learners have trouble following directions; 6) problems interact; 7) interface features are not obvious to learners; 8) “help” does not always help. After articulating these problems, Mack et al. discuss possible cures, pointing out that unaided self-study is not appropriate for novices to learn word processing. People are reluctant to read thick volumes before starting, and they become too passive when following tutorials. This article, when considered with Carroll and Mack (1984), articulates the major assumptions about how novices learn that drive the research at IBM Watson.

The question of why some people have more difficulty than others in learning word-processing programs was posed by Gomez, Egan, and Bowers (1986). In several studies, they found that older people had more trouble than younger people and that people with poor spatial memory had more trouble than people with good spatial memory. These correlations were stable over a variety of conditions (the amount of practice time, the type of terminal, and the specific editing tasks) and in relation to other characteristics (education, reasoning ability, and associative memory ability). The authors suggest that the characteristics of users need to be more thoroughly planned for in system design.

Two major learning models have been produced for word processing. Card, Moran, and Newell (1983) first proposed GOMS (a family of models that stand for Goals-Operators-Methods-Selection Rules). Actually, GOMS is a model of human interaction with computers, but it was developed using text-editing programs. Consider GOMS’ explanation of how to move text. The user normally has a goal in mind when highlighting text and inserting it elsewhere. The user also has a number of commands or operators that can be used to move the text. Further, the operators have methods that are used to carry out the task. The user selects the strategy for accomplishing the goal in mind. GOMS has been central to model development in human-computer interaction because the authors have had success predicting how long it would take a person to reach some of the goals.

A second major model comes from Singley and Anderson (1987-1988). They advance the work done in GOMS, and in the cognitive complexity theory of Kieras and Polson (1985), by building a model of how people learn a first line editor and how they then transfer the knowledge to a new line editor and to a screen editor. Singley and Anderson find that, although novices may be slowed by a lack of knowledge (e.g., the person commands the computer, not vice versa, and the computer does not recognize and correct mistakes), they quickly get past those notions and on to the business of learning the particular procedures of the editor. Singley and Anderson found that people transferred local and task-specific knowledge; that is, they did not find general strategies for transfer at work for the learners who moved to a new editor. They also found little interference to new procedures. This study should spark more research into long-term learning.

Goal of Improving User Interface Design

Those who study interface design are interested in solving user problems by changing the software itself. Typically, people studying interface design are asking questions such as these: Is there a best style of interface? How rigidly do we have to follow conventions? What kinds of markers should we use? Will it help people to see the structure of the task? The field of interface design has traditionally favored artistic answers to these questions. So, researchers who study interface design face a difficult battle when they argue that users know better than the artists.

Whiteside, Jones, Levy, and Wixon (1985) provide a good example of research into interface style, even though they focus on operating systems rather than on word-processing programs per se. In their research, they evaluate seven systems that display three types of interface style: command, menu, and iconic. They find no clear-cut style that is best for all new users. They had expected the menu style to be best for new users, but actually the new users performed worst on the menu system, and comparably better on command and iconic systems. Although the Whiteside et al. study is not thorough enough to conclusively decry menus, it points out that a style we think users will prefer is not adequate to make up for other problems in the interface. They conclude, and rightfully so, that the interface’s usefulness is more than its style.

Gardiner and Christie’s Applying Cognitive Psychology to User Interface Design (1987) is an important book because it takes research findings to the artists who design systems. Those artists live by guidelines, so Gardiner and Christie give them research-based guidelines. Take the case of metaphors and analogies. Designers use both, but they often try to suggest one-to-one correspondence. This book makes clearer how to use analogies and metaphors when it specifies that such devices should be in common usage, should provide information about boundaries, should show the essential characteristics, and should make explicit the nature of the mismatches...
found in the analogy/metaphor (pp. 230-231). The suggestions are consonant with the Mack et al. (1983) findings about problems that metaphors pose to new users of word processing.

**Goal of Evaluating and Developing New Products**

Product evaluation work aimed at word processing comes in two varieties: academic and commercial. The academic work tries to test emerging features of a class of products (e.g., Gould's 1981 work on the importance of cursor speed to word processing) and also to establish standard ways of testing both features and whole products (in our case, editing and word-processing programs). This effort normally runs both ahead of and behind the consumer technology, setting its clock to product development. The commercial work, found in such magazines as *Byte, InfoWorld, MacWorld,* and *PC,* develops critiques by taking the major products on the market, submitting them to comparative tests, and then publishing the results. The commercial work serves as a consumer report, while the academic work evaluates in order to gain insights for development.

The Roberts study (Roberts, 1979; Roberts & Moran, 1983), growing out of Roberts' dissertation, provides a good example of academic evaluation. It proposed a standardized evaluation for text editors, suggesting 212 editing tasks that potentially can be performed, and a small set of typical tasks considered to be the most common editing tasks. The work aimed to develop a standard method for testing editing and word-processing programs; and, even though it received substantial criticism (e.g., Borenstein, 1985), the study energized the thinking about editing software. After its appearance, it became more common for authors to consider the typical tasks to be performed as a reasonable basis for feature-based or task-based evaluation.

Another typical study is Good's (1985). He looked at keystroke records from five word-processing programs already in use and was able to take the records to build a set of commands for a new editing program that included all the powerful and frequently used features of the five programs studied. Commands that fared less well were analyzed further for their power, and the system was developed based on the feedback from writers that Good received unobtrusively through his keystroke record program.

**Questions the Two Groups Might Share**

The brief research summaries above suggest interests we share with researchers in human-computer interaction. Even though we do not share an interest in the quality of writing, we do share curiosities about learning and using word-processing programs and about the development of word-processing technology. Indeed, we might pose a number of common questions.

*How quickly and easily do people learn word-processing programs?*

In general, human-computer interaction researchers have found learning how to use word-processing software more difficult than they expected. It is true that their studies of learning tend to be short-term (mimicking the one- or two-day training session). But these studies consistently find that novices have a poor understanding of the computer and that they make many errors.

Consider, for a moment, the findings as teachers of college students. The training and learning studies are usually run using office workers, and hence may not translate to today's more sophisticated college students. But then again, we may encounter similar learning problems connected to new or complex writing systems (e.g., desktop publishing features). If we expect that students know computers and word processing when they start our classes, or that they learn them quickly and easily, what kinds of stress do we put on students who do not learn them quickly and easily? Our classrooms, and by extension our research, could profit from our paying attention to the features of word processing that students learn quickly, or not at all. Even though paying direct attention to word processing makes us vulnerable to the charge of "teaching technology rather than writing," if such direct attention is briefly given, it may pay off in the long run. We do not want problems with the mechanics of the technology to inhibit the learning of writing, and we do not want to underestimate the students' abilities, either.

*How can the learning of word processing be enabled?*

The question of how to encourage and enable learning is a lively one. There are many reasons to think that much work is left to be done. Almost all the work at IBM Watson, for example, is aimed at enabling learning. Researchers have pursued strategies for encouraging learners to be active, to explore, and have also developed training materials that enable and guide. But the recent work of Singley and Anderson (1987-1988) may challenge the Watson approach, as they suggest that, when people "get down to the business of learning," they focus on the procedures and soon no longer need the types of guidance being developed in the minimalist approach.

When we add into the equation our interests in developing good habits for writing and for using word processing to enable the production of quality text, then the question becomes even more lively and less settled.

*How does the relationship between person and machine change over time and with use?*

This question, which is essential to educators, has only begun to attract attention in human-computer interaction research. The transfer studies, which track what happens when a person learns a new word-processing program, begin to phrase the question; but long-term studies are not the norm. Singley and Anderson (1987-1988) is one of the longer recent studies; it took place over six intensive days and involved experienced typists who were typing and editing manuscripts. The human-computer interaction researchers tend to favor laboratory settings for configuring their research, and they do not have the easy and prolonged contact with writers that we teachers have. In addition, Hawisher (1988), in her review of research studies in computers and composition, has pointed out that little long-term research has been done in composition studies, as well, although our studies would be considered long-term in relation to most human-computer interaction work. This question of long-term learning is one that is surfacing simultaneously in both groups, and the divergent approaches could lead to exciting findings and disputes.

*Will we eventually develop a "best" interface or an "ideal" word-processing technology?*

Human-computer interaction researchers are always asking this question and never answering it. It doesn't make sense that one
Human-computer interaction researchers have not posed questions in the frames of learning to write or of facilitating writing habits. If quality of prose or quality of composing process were important to the evaluations in that field, some of their conclusions might be different. Take, as an example, desktop publishing. PAGEMAKER is consistently judged superior, but it makes laying out a technical manual arduous. PAGEMAKER is harder to use than other programs (such asVENTURA or READYSETGO) that give control over the precise placement of text. My point is two-edged: first, that human-computer interaction needs to critique out of a base of writing theory and writing process as well as out of a base of technological sophistication; second, that teaching and research using word processing need to incorporate the attitudes that look for the best and that critique the features and programs in use.

Conclusion

The work in human-computer interaction can help us think more carefully about the characteristics of learning word processing and of using particular word-processing programs. But it does not give us answers. Even though we can articulate questions of interest to both fields, the researchers in human-computer interaction do not have the answers to our research questions because their research does not focus on writing process or writing quality. Indeed, they could profit from a better understanding of how the writing task and the writer's skill interact with the person learning to use a word-processing program. Such an understanding would deepen their work: They could focus their evaluations on quality of product as well as on efficiency of using the procedures.

A knowledge of work in human-computer interaction can help us with a major question underlying many studies: How much of the change in writing habit is due to the technology itself? Currently, that question is normally intertwined with the question of teaching method and milieu. A better understanding of the literature in human-computer interaction can help us sort out how the technology itself interacts with the writing process because this literature has more precise and workable ways of talking about the functioning components of the technology. Studies like Whiteside's and Roberts' give us ways to think about whether differences between a study using WORDSTAR and a study usingMACWRITE can be thought of as differences arising out of differences in interface style and program complexity. Such an injection of reasoning about technology can certainly aid us in the work of sorting out how computers influence the teaching and learning of writing.

Patricia Sullivan teaches at Purdue University in West Lafayette, Indiana.

Notes

1. I thank James Porter for his helpful reading of this text.

2. A caution: This discussion does not pretend to catalog every study of word processing, and it does not focus on aspects of human-computer interaction other than word processing. A comprehensive study of that field would show, for example, that the study of how people search for information, another topic studied in human-computer interaction, could shed light on the process of doing library research. Exploring all points of convergence is beyond the scope of this paper.

3. Two reading plans make sense: exploring goals and issues, or understanding one total approach. This paper suggests issues and studies connected to those goals and issues. For people interested in a particular goal or issue, the studies listed in Table 1 can serve as a guide. For people more interested in exploring a coordinated position, the studies coming from the IBM Watson Research Center can serve as a guide. The researchers at Watson (people who have been authors on more than one article include Carroll, Carrithers, Gould, Lewis, Mack, Rosson, and Thomas) demonstrate what can be accomplished when a research group turns its coordinated attention to word processing.

References


Walker, N., & Olson, J. R. (1988). Designing keybindings to be easy to learn and resistant to forgetting even when the set of commands is large. *CHI ’88 Proceedings*, 201-206.


**Appendix**

**Resources for Further Reading**

**Journals/Periodicals**

- *Human Factors* (1959- )
- *SIGCHI Bulletin* (1982- )
- *Behaviour and Information Technology* (1982- )
- *Human-Computer Interaction* (1985- )

**Conferences/Proceedings**

- Human Factors Society Annual Meeting (1957- )
  Proceedings published by The Human Factors Society (P.O. Box 1369, Santa Monica, CA 90406)

- ACM CHI Conference (1982- )
  Sponsored by the Special Interest Group on Computers and Human Interaction (SIGCHI) of The Association for Computing Machinery (ACM), with the proceedings a special issue of *SIGCHI Bulletin* (11 W. 42nd St., New York, NY 10036)

- IFIP Interact Conference (1984- )
  Sponsored bi-yearly by the International Federation of Information Processing, with proceedings published by North-Holland.
Table 1: Sampling of HCI Studies on Word-Processing Programs

<table>
<thead>
<tr>
<th>Goals</th>
<th>Authors</th>
<th>Issues</th>
</tr>
</thead>
<tbody>
<tr>
<td>Training</td>
<td>Carroll &amp; Carrithers (1984)</td>
<td>limit the interface choices to guide new users</td>
</tr>
<tr>
<td></td>
<td>Carroll (1984)</td>
<td>compare minimalist/traditional training</td>
</tr>
<tr>
<td></td>
<td>Carroll et al. (1986)</td>
<td>explore training/learning</td>
</tr>
<tr>
<td></td>
<td>Carroll &amp; Thomas (1982)</td>
<td>argue for metaphors in learning</td>
</tr>
<tr>
<td></td>
<td>Pope (1985)</td>
<td>count where users spend time</td>
</tr>
<tr>
<td></td>
<td>Czaja et al. (1986)</td>
<td>compare computer, book and stand-up training methods</td>
</tr>
<tr>
<td></td>
<td>Raban (1988)</td>
<td>compare guided exploration and instruction</td>
</tr>
</tbody>
</table>

Learning (Skill Acquisition)

<table>
<thead>
<tr>
<th>Goals</th>
<th>Authors</th>
<th>Issues</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Carroll &amp; Mack (1983)</td>
<td>observe for active learning</td>
</tr>
<tr>
<td></td>
<td>Folley &amp; Willege (1982)</td>
<td>show experts/novices learn differently</td>
</tr>
<tr>
<td></td>
<td>Mack, Lewis &amp; Carroll (1983)</td>
<td>observe novices and articulate problems</td>
</tr>
<tr>
<td></td>
<td>Allen (1982)</td>
<td>observe actual use problems</td>
</tr>
<tr>
<td></td>
<td>Foltz et al. (1988)</td>
<td>study transfer of menu knowledge to a new word-processing program</td>
</tr>
<tr>
<td></td>
<td>Gomez, Egan &amp; Bowers (1986)</td>
<td>find types of people apt to learn faster</td>
</tr>
<tr>
<td></td>
<td>van Muylwijsk et al. (1987)</td>
<td>assert assumptions about user variability</td>
</tr>
<tr>
<td>Learning (Models)</td>
<td>Card, Moran &amp; Newell (1980)</td>
<td>assert GOMS model of learning editing</td>
</tr>
<tr>
<td>---------------------------</td>
<td>-----------------------------</td>
<td>-------------------------------------</td>
</tr>
<tr>
<td></td>
<td>Card et al. (1984)</td>
<td>apply GOMS to actual tasks</td>
</tr>
<tr>
<td></td>
<td>Polson &amp; Kieras (1985); Kieras &amp; Polson (1985)</td>
<td>assert cognitive complexity model of learning</td>
</tr>
<tr>
<td></td>
<td>Singley &amp; Anderson (1985, 1987-1988)</td>
<td>assert model for transfer of learning to a new editor or word-processing program</td>
</tr>
<tr>
<td>User Interface Design</td>
<td>Whiteside et al. (1985)</td>
<td>evaluate interface styles (command menu and iconic)</td>
</tr>
<tr>
<td></td>
<td>Mack (1985)</td>
<td>propose/design an interface for new users</td>
</tr>
<tr>
<td></td>
<td>Laurel (1986)</td>
<td>chart subjective nature of experience</td>
</tr>
<tr>
<td></td>
<td>Walker &amp; Olson (1988)</td>
<td>develop rules for keybinding</td>
</tr>
<tr>
<td></td>
<td>Carroll &amp; Kay (1985)</td>
<td>describe explorer interface</td>
</tr>
<tr>
<td></td>
<td>Gardiner &amp; Christie (1987)</td>
<td>present psychological backing for guidelines</td>
</tr>
<tr>
<td></td>
<td>Kindborg &amp; Kollerbauer (1987)</td>
<td>analyze visual languages</td>
</tr>
<tr>
<td>Product Evaluation</td>
<td>many unnamed for magazines like Byte, PC, InfoWorld, Seybold Reports</td>
<td>present comparative evaluations of products’ performance and features</td>
</tr>
<tr>
<td></td>
<td>Embley &amp; Nagy (1981)</td>
<td>review research on editors to 1980</td>
</tr>
<tr>
<td></td>
<td>Roberts (1979); Roberts &amp; Moran (1983)</td>
<td>develop a standardized set of evaluations for features</td>
</tr>
<tr>
<td></td>
<td>Feruta, Scofield, &amp; Shaw (1982)</td>
<td>survey formatting techniques</td>
</tr>
<tr>
<td></td>
<td>Borenstein (1985)</td>
<td>critique Roberts and Moran study</td>
</tr>
<tr>
<td></td>
<td>Meyrowitz &amp; van Dam (1982 a b)</td>
<td>survey interfaces focusing on design and functionality, a bit on usability</td>
</tr>
<tr>
<td>Product Development</td>
<td>Ferm et al. (1987)</td>
<td>explore mix of graphics and text features</td>
</tr>
<tr>
<td>---------------------</td>
<td>-------------------</td>
<td>------------------------------------------</td>
</tr>
<tr>
<td></td>
<td>Thomas (1987)</td>
<td>explore graphical features</td>
</tr>
<tr>
<td></td>
<td>Gould, Lewis, &amp; Barmes (1985)</td>
<td>study impact of cursor speed</td>
</tr>
<tr>
<td></td>
<td>Good (1985)</td>
<td>analyze keystroke record for commands' use</td>
</tr>
<tr>
<td></td>
<td>Teubner &amp; Vaske (1988)</td>
<td>develop monitoring techniques for office research</td>
</tr>
</tbody>
</table>