WARRANT - A Learner-Centered Computer Environment For Critical Reading, Reasoning, and Writing

Preston K. Covey
Carnegie Mellon University, dtrollcovey@gmail.com

David S. Kaufer
Carnegie Mellon University, kaufer@andrew.cmu.edu

Christine Neuworth
Carnegie Mellon University

Follow this and additional works at: http://repository.cmu.edu/philosophy

Part of the Philosophy Commons
WARRANT

- A Learner-Centered Computer Environment -

For Critical Reading, Reasoning, and Writing

Preston Covey, Department of Philosophy
David Kaufer, Department of English
Christine Neuwirth, Communications Design Center

Carnegie-Mellon University

April 8, 1984
Abstract

The problem of literacy is not a writing problem alone. Skills of reading, writing, and reasoning are interactive and interdependent. Nonetheless, instruction in writing at the post-secondary level is seldom integrated with instruction in reading or reasoning. The obstacles facing such an integration are twofold: (1) We don't have enough knowledge of reading, writing, and reasoning processes in specific task domains to integrate instruction; (2) We have not, until recently, had powerful enough technologies to deliver instruction as flexibly as students need it in the course of their reading and writing.

We propose to develop a computer system, WARRANT (Writing, And Reasoned Reading, About Normative Texts) that addresses both these problems. Built on a technology that will reach the college classroom in 2-5 years, WARRANT provides a learner-centered, integrated task environment and flexible guidance for critical reading, reasoning, and writing.

WARRANT combines the power of heuristic learning with learning by feedback from models. WARRANT will offer learners not only explicit strategies for working through a particular reading/reasoning-writing task, but funds of information about how other persons (both novices and experts) applied these strategies (with varying degrees of success) when working through the same task.

The outcomes of our project turn on viewing WARRANT either as a teacher capable of structuring effective learning environments or viewing it as a laboratory that will allow teachers to experiment with different learning environments to see which are effective, which not. Viewing WARRANT as a teacher, Covey and Kaufer will write a text that details effective learning environments for teaching normative reasoning, reading, and writing. Viewing WARRANT as a laboratory, Neuwirth, Kaufer, and Geisler, will write a book for teachers that details how WARRANT can be modified to test the effectiveness of different learning environments.

We plan to use the WARRANT system first in selected pilot courses, and then in our freshman core curriculum courses in composition and philosophic analysis. Formative evaluation of the system and its pedagogy is built into our protocol research agenda.
Table of Contents

1. The Problem 1

2. The Project 3

2.1. Our Approach to the Literacy Problem 3

2.1.1. Literacy and Empirics 3

2.1.2. Literacy and Teaching Styles 3

2.1.3. Literacy and Computers 4

2.1.4. Distinguishing Our Approach from Final Solutions 5

2.1.5. WARRANT as Teacher and as Experimental Lab 5

2.1.6. The Task Domain of WARRANT: Normative Argument 6

2.2. Guide to the Appendixes 6

2.3. Our Work Plan 7

2.3.1. Systems Development 7

2.3.2. Protocol Research 7

2.4. Statement of Intended Outcomes 8

2.5. Capacity and Commitment for Completing This Project 9

2.6. Discussion of Plans for Wider Impact 10

3. Budget Summary Sheets 13

4. Budget Detail and Narrative 15

4.1. DIRECT COSTS 15

4.1.1. Salaries and Wages 15

4.1.1.1. Salary and Wages Narrative 16

4.1.2. Employee Benefits 18

4.1.2.1. Benefits Narrative 18

4.1.3. Travel 19

4.1.3.1. Travel Narrative 19

4.1.4. Equipment 20

4.1.4.1. Equipment Narrative 20

4.1.5. Materials and Supplies 21

4.1.5.1. Materials Narrative 22

4.1.6. Consultants (and Research Assistants) 22

4.1.6.1. Consultant (and Research Assistant) Narrative 24

4.1.7. Other 24

4.1.7.1. Narrative 25

4.1.8. Total Direct Costs 25

4.1.9. Modified Total Direct Costs (MTDC) 26

4.2. INDIRECT COSTS 26

4.3. TOTAL PROJECT COSTS 26

5. APPENDIX I. 29

WARRANT: Pedagogy and Curriculum Testing 29

5.1. The Educational Theory that Informs the WARRANT System 29

5.1.1. A Process Model of Skill Acquisition 29

5.1.1.1. Goals 30

5.1.1.2. Strategies 30

5.1.1.3. Discriminations 30

5.1.1.4. Scheduling goals 30

5.1.2. Learning by Doing and Discovery Learning 31
5.1.2.1. Structured Learning
5.1.2.2. Freedom with Structure

5.2. The Educational Theory that WARRANT Can Inform
5.2.0.1. The Traditional Problem of Curriculum Testing
5.2.0.2. WARRANT as an Environment to Test Curricula

5.3. Curricular Materials to Be Used and Tested on WARRANT
5.3.1. Readings
5.3.2. Goals
5.3.3. Strategies/Heuristics
5.3.4. Teaching Discrimination Through Expert and Novice Models
5.3.5. Teaching Discrimination Through Assessments of Models

5.4. The Research to Warrant WARRANT
5.4.1. Methodology
5.4.2. Our Selection of Readings
5.4.3. Our Selection of Models
5.4.4. Phases of Our Protocol Research
5.4.4.1. The First Phase
5.4.4.2. The Second Phase
5.4.4.3. The Third Phase

5.5. A Hypothetical Interaction Session
5.5.1. Some Qualifications and Caveats

6. APPENDIX II.

WARRANT: System Design and Implementation

6.1. Our Design Goals
6.1.1. Flexible Structure
6.1.2. Integration of Tools
6.1.3. Easy to learn and to use

6.2. System Architecture
6.2.1. System Hardware
6.2.1.1. Improved Screen Readability
6.2.1.2. Multiple Windowing
6.2.1.3. What You See Is What You Get Editing

6.2.2. System Software
6.2.2.1. The text-editor/window-management program.

6.2.3. Why We Chose this System Hardware and Software
6.2.3.1. WARRANT’s requirements
6.2.3.2. Building on Existing Tools
6.2.3.3. Increased Likelihood for Use on Campus
6.2.3.4. Good Potential for Dissemination

6.3. Our Proposed Program
6.3.1. Design Methodology
6.3.1.1. Task Analysis
6.3.1.2. Formative Evaluation

6.3.2. Tools for Reading and Reasoning
6.3.2.1. On-line Reading: Interactive Texts
6.3.2.2. Prompts: Modelling Goals and Strategies for Reasoning
6.3.2.3. Note Cards: Integrating Reading and Writing
6.3.2.4. Monitoring: Modelling When to Pursue a Goal
6.3.2.5. Do Goals and Strategies Generalize to Other Texts?
6.3.2.6. Limitations and Long-range Goals  63
6.3.3. Tools for Writing & Reasoning  64
   6.3.3.1. Planning  64
   6.3.3.2. Organizing  64
   6.3.3.3. Revising  65
6.3.4. Tools for Managing the Environment  65
   6.3.4.1. Curriculum Guides  65
   6.3.4.2. Curricular Prerequisites  66
   6.3.4.3. Statistics Gathering  66
6.4. Plans for Formal Evaluations  67
7. APPENDIX III.  69
Project Commitment and Support Documentation  69
   7.1. Institutional Support Units  69
   7.2. Project Personnel  73
   7.3. VITAE of Project Personnel  75
# List of Figures

| Figure 5-1: | Janice's first encounter with WARRANT | 40 |
| Figure 5-2: | The menu of reading selections on reverse discrimination. | 41 |
| Figure 5-3: | The first few paragraphs of the Shiner article. | 42 |
| Figure 5-4: | Help menu for the first few paragraphs of the Shiner article. | 43 |
| Figure 5-5: | A Menu of relevant goals for the first paragraph of the Shiner article. | 43 |
| Figure 5-6: | An explanation of an author's opening moves. | 44 |
| Figure 5-7: | Strategies for identifying the generally-held position. | 45 |
| Figure 5-8: | Janice makes a note on the generally-held position. | 45 |
| Figure 5-9: | Subsequent entries into WARRANT. | 46 |
| Figure 5-10: | The set of tasks for completing a critical reading, reasoning and writing assignment. | 47 |
1. The Problem

A direct implication of the national literacy crisis is that college students, even at "better" institutions, do not write well. Colleges and universities are responding by establishing remedial programs, by hiring more faculty to teach composition, and by funding cross-curricular writing programs. These responses, however apt, address symptoms. The real problem with students' inability to write well goes beyond writing. It lies in their inability to read critically and, more deeply, in their inability to reason critically about what they read or wish to write about. This problem is epidemic, it cuts across disciplines, and it extends well beyond academic life into business and the professions.

The problem of literacy is hardly a writing problem alone. Skills of reading, writing, and reasoning are highly interactive and interdependent. Reading skill is demonstrated and evaluated through writing. Writing skill is informed and inspired through reading. Reasoning skill is shaped and tempered through reading; demonstrated and evaluated through writing; presupposed and exercised in both. So, clearly, the problem of literacy is multi-dimensional, cross-disciplinary and circular: reading courses must be writing courses must be reasoning courses; and vice versa. This conception of the problem is commonsense if not commonplace. But solutions will require uncommon means.

Epidemic illiteracy and its causes aside, there are severe problems endemic to teaching critical reading, writing, or reasoning alone — let alone in some coordinated fashion. One is the problem of feedback: students invariably need much more feedback on their reading, writing, and reasoning than teachers can possibly anticipate—or deliver. Another is the problem of practice: Students can’t simply be expected to commit a set of techniques to memory; they need repeated practice—practice applying these techniques, practice evaluating the results of these applications, and practice generating alternative strategies when a given technique seems not to apply. We cannot provide the kinds or amount of practice required—let alone feedback on it all—by usual means.

Worse, what feedback can be provided typically comes long after the student can most effectively make use of it, which would be—optimally—in the very process of reading, reasoning, or writing. Indeed, practice and feedback of the usual sort (evaluated assignments) are less appropriate for enhancing skills than certifying their
presence or absence. Grades and comments are usually addressed to the qualities of some essay or examination, some product of the student's efforts. Those efforts themselves generally take place in untutored isolation, by a process the teacher can only guess but hardly help.

The alternative is clear, if difficult to offer: some guidance in how to proceed, some clues about what to do, in the very process of reading, reasoning, or writing. If we are trying to teach judgment and skill in how to do something, we need to provide guidance in the very doing of it—not merely commentary on the outcome or advice for prospective application.

Unfortunately, it is not humanly possible, much less cost-effective, for us to monitor, with any frequency, how students actually go about applying the techniques we give them—or to be there to suggest to them alternative paths when current ones fail. Typically, when students have the opportunity to ask our advice (in an office or classroom) about how to proceed in some reading, reasoning, or writing task, there is typically a lag between their noting the advice and trying to apply it. By contrast, the advantages of immediate feedback and application are well known.

It is the difficulty of effectively teaching actual skills—and the special difficulty and necessity of coordinating the teaching of critical reading, reasoning, and writing—that we wish especially to acknowledge in our approach. It is this difficulty that demands an uncommon approach, beyond the competence of unaided pedagogical savvy and traditional practice— an uncommon emphasis on the processes as well as the products of learning and application, and, consequently, exploitation of novel research models and technology that are process- and learner-oriented.
2. The Project

2.1. Our Approach to the Literacy Problem

2.1.1. Literacy and Empirics

The approach we take to the literacy problem is based on a number of assumptions. First we assume that the problem stems in part for want of empirical knowledge about what people actually do when they read, write, and reason. This assumption guides the thinking of those educators who suggest that our prescriptions for what students should do must be firmly rooted in what they do and can do.

We think it behooves any teacher to understand the thought processes that students go through when trying to perform some target task(s) in reading, reasoning, and writing—especially when their performance is unsatisfactory—in order to diagnose their difficulties and design better learning aids. There is a well-established technique for gaining access to the cognitive processes of either experts or novices as they go about various problem-solving tasks. It consists in having them "think out loud" and recording their reports of the steps they take. These protocols are then analyzed for common patterns that seem to contribute to success or failure in the tasks in question.

2.1.2. Literacy and Teaching Styles

A second assumption behind our approach to the literacy problem is that, given the little we do know—through protocol data, and retrospective and introspective accounts—about the processes of writers and readers, we should hardly find it surprising that careful writing and reading are difficult skills to learn.

Studies in composing/reading suggest that (1) writing and reading processes are recursive, meaning that they call one another back and forth; (2) writing and reading processes are heterogeneous, meaning that they can range anywhere from skill of inquiry and social perception to skills of word recognition and the perception of meter; (3) writing and reading processes are hierarchical, meaning that they are radically decomposable into tasks and subtasks and these subtasks, into even more subtasks. Unlike simpler skills that can be learned in a linear schedule with a rigid agenda of independent and homogeneous goals (e.g., tying one's shoe, adding numbers, spelling words), reading and writing need to be learned in a rich environment where everything, in Peter Elbow's term, must be "cooking."
The empirical considerations that help to explain the difficulty of learning to read and write also help to explain the difficulties teachers face when they try to teach processes of reading and writing. Teachers can scarcely monitor, much less articulate, all there is for students to learn. They cannot, through a lecture format, lead students step-by-step through a process that defies linearity. How then do effective teachers of reading and writing teach?

In a study he conducted to assess different styles of teaching reasoning and writing, Hillocks\(^3\) found that the most effective teachers seem to be those who are able to structure an environment that makes it possible for students to learn to read, reason, and write on their own initiative and out of their own curiosity.

If we could isolate, package, and market the "learning environments" that master teachers create, we would go a long way toward solving the literacy problem. Unfortunately, even when master teachers write about what they do in the classroom, they can only report—not reconstruct—what they do. They can describe the firm or shaky paths taken by their students, but they can't reconstruct, for purposes of experimentation and dissemination, how they manage to keep a student on promising paths even after that student falters.

2.1.3. Literacy and Computers

These considerations lead us to a third assumption behind our approach, namely that computers, when sufficiently powerful, can be effective at structuring learning environments for processes that are at once recursive, heterogeneous, and hierarchical. The disappointment over the first generation of computer-based writing aids was that they lacked the power (viz., speed and memory) that would have enabled them to structure such environments.

All such programming aids could do was imitate bad teachers. They were nonadaptive, inflexible, and too often condescending. They allowed the student no freedom to take control over his or her own processes. Instead of intervening only when the student lost control, they intervened incessantly and without provocation. They did not help situate the student within a rich social environment of other writers and readers; instead, they isolated the student from other authors and readers. They represented writing as divorced from reading and reasoning, and as a linear and homogeneous process rather than a hierarchical and heterogeneous one.

As we mentioned above, the designers of this first generation of writing aids were
greatly constrained by technology. It wasn’t so much that they wanted to create “bad teachers.” It’s just that a “bad teacher” was all a reasonably priced educational package could be. Times are changing. Granted, we are a long way from developing a computer that has the intelligence (and responsiveness) of a master teacher. But computers are quickly becoming cheap enough and powerful enough to make it feasible for them to provide the kind of structured “learning environment” that we associate with master teachers. WARRANT, the computer system we propose to build, represents the first of a new generation of “learning environments” for students writing, reading, and reasoning.

2.1.4. Distinguishing Our Approach from Final Solutions

It is important to remember that our approach to the literacy problem is just that—an approach without, as yet, clear answers. An approach that urges the empirical study of writing and reading processes does not yet say what reading and writing processes go awry when a student lapses. An approach that urges the structuring of learning environments for teaching reading and writing does not yet say what learning environments are really effective. Nor can an approach urging these learning environments to develop on powerful computers tell us anything, by itself, about what these effective environments really are.

The Ambiguity of Illiteracy

Readers should not miss the ambiguity when we speak of WARRANT structuring “learning environments” for students’ writing, reading, and reasoning. Is a “learning environment” meant to be an effort on the teacher’s part or an achievement on the student’s? This ambiguity strikes at the very heart of the literacy problem. Does illiteracy betray the ignorance of the student who can’t take advantage of a “learning environment” or of the teacher who can’t create one? Teachers of reading, reasoning, and writing characteristically expect skilled performances, but they just as typically do not know what their expectations come to nor do they have a clear or articulate grasp of the requirements on students their expectations impose.

2.1.5. WARRANT as Teacher and as Experimental Lab

Our WARRANT project exploits the ambiguity between using technology to teach students how to read and write effectively vs. experimenting with technology to teach teachers how to teach effectively. We shall develop WARRANT with these different, but interactive, goals in mind. That is, we shall develop WARRANT partly
to see how well it can structure effective writing and reading environments for students. But we shall also develop it to see how well it can help teachers assess the effectiveness of different learning environments.

2.1.6. The Task Domain of WARRANT: Normative Argument

There are all manner of reading, reasoning, and writing tasks—and no omnibus approach for teaching skill in every variety and application. The feasibility of any project like ours demands focus, specificity of target tasks and skills.

We will focus on critical reading and reasoning of the sort required for analytical or argumentative writing. We will emphasize skills needed to address normative issues that anyone is bound (by duty or circumstance) to wrestle with in private, civic, or professional life. Value-laden issues provide as good a crucible for the Word as any. We assume (and will argue) that the tools of critical inquiry and argument required in reading and writing about, say, social value and policy issues are quite generic—and indispensable to educated persons in all manner of reading, reasoning, and writing tasks.

Normative issues demand special sensitivity to audience, special consideration of the social context in which one is writing. WARRANT will provide a social environment, in effect, a dialectic of viewpoints, to reinforce the sensitivity to audience and context required by normative inquiry and argument.

2.2. Guide to the Appendixes

The WARRANT project comprises activities that are necessarily multi-disciplinary, encompassing empirical, pedagogical, and computational work. The scope of the project has persuaded us to include appendixes that go into greater depth than a general description allows. Appendix I details the pedagogical and empirical assumptions and methods underlying the WARRANT system. Appendix II details the design specifications for WARRANT, which have been thoroughly reviewed and approved by a member of the computer science department at Carnegie-Mellon (Donald McCracken). McCracken will serve as our technical consultant for the project.

2.3. Our Work Plan

Since we want to build WARRANT both to train students and to train teachers, and since WARRANT itself represents a complex technology, we will have two highly symbiotic activities going on during every phase of the three year project. For
convenience, we list each of these activities separately:

2.3.1. Systems Development

We call the first activity "systems development" because it is concerned with WARRANT from the computer science end of things. WARRANT represents the merging of three sophisticated technologies: A bit-mapped display text-editor with high resolution graphics and multiple-windowing capabilities (i.e., like the APPLE Macintosh, but more powerful); a prototype personal computing system with twice the power and four times the speed of the IBM PC; and a powerful information management network modelled after the ZOG system. (These are carefully detailed in appendix 2).

During the first year of the project, the systems development team will concentrate its efforts to make WARRANT a functioning, integrated, computer system. By the end of the first year, the team will begin story-boarding and they will build detailed mock-ups of possible interfaces with students. At the beginning of the second year, they will begin to code information about the normative reading/writing task into WARRANT's data-base and they will take formative evaluations of students' interacting with limited parts of the WARRANT system. By the end of the second year, the full data-base will be coded and more general formative evaluations will be conducted. In the third year, the system development team will develop dynamic and log statistics facilities that will allow us to keep automatic records of students' progress through the system.

2.3.2. Protocol Research

We call the second activity "protocol research" because it comprises the empirical phase of the project. Given our educational philosophy (see appendix 1), we plan that students using WARRANT will be able to tap both expert and novice goals and strategies, and that they will be able to access expert assessments of the efficacy of these goals and strategies. The protocol research team will thus have the responsibility of collecting protocols from groups of expert and novice writers, and leading discussions of experts assessing the efficacy of the goals and strategies that they and other writers used during the task. The protocol research team will work with the PIs in analyzing the protocols.

In the first year, the head research assistant (Geisler) and two graduate students will initially run a protocol on approximately 5 expert writers working through the
normative reading/writing task. Geisler and the Pis will then analyze the protocols, seeking to distill expert goals, strategies, and paths through the task. They will then take protocols of approximately 5 novice writers working through the task and they will use this distilled information as a way of helping to structure novices' paths through the task before they embark on it. From these protocols, the protocol research team and the Pis will try to formulate lists of novice goals and strategies in the manner of their analysis of the experts.

During the end of the first and the second year, the protocol research team will be working actively with the systems development team to code the expert and novice strategies into WARRANT's data base. During the second year as well, the protocol research team, having carefully collated and edited the expert and novice protocols from the previous year, will bring back those who served as experts and have them describe, evaluate, and rank the goals and strategies they perceive in the protocols for their efficacy. The protocol research team will then work with the system development team in coding these expert assessments for WARRANT's data-base. In the third and final year, the protocol research team will conduct a full-scale formative evaluation on approximately 20 students using WARRANT.

2.4. Statement of Intended Outcomes

Within the duration of the project, we plan four major outcomes. First, the WARRANT system will be operational and will have undergone a major formative evaluation on one reading/writing task in the domain of normative argument. Second, Covey and Kaufer, working with the expert and novice models, will have substantially completed a text for students on goals and strategies that are effective for reasoning about normative issues. Third, Neuwirth, Kaufer, and Geisler, applying their experience at using WARRANT as an effective teaching tool, will write a book for teachers that describes how they used WARRANT as a tool to test different pedagogies. Fourth, we plan that the technology underlying WARRANT (if not exactly the WARRANT system itself) will be readily assessible (at least to) the colleges and universities that are members of the Inter-University Consortium for Educational Computing (ICEC). See Appendix III for the names of these institutions.

We are also planning two long range outcomes for WARRANT. Within 2-5 years beyond the expiration of the project, we intend to make WARRANT available to teachers at other institutions. This goal will easily be realized as the new generation of personal computers and computing technology becomes widely affordable and
available to colleges and universities even with modest computer budgets.

We also plan to build an authoring system for WARRANT as a follow-up to the current project so that teachers at other institutions without a programming background can freely modify and experiment with the learning environment it structures.

2.5. Capacity and Commitment for Completing This Project

To confirm WARRANT's hardware requirements, feasibility of implementation, and timetable-budget, we have consulted the following people: Donald McCracken, Research Computer Scientist, Department of Computer Science, Carnegie-Mellon University; James Gosling, System Designer, Information Technology Center, Carnegie-Mellon University; and Robert Ramey, Senior Scientist, Engineering and Applied Sciences, Mellon Institute, Carnegie-Mellon University (whom we have recruited prospectively as our chief systems programmer for the project).

Donald McCracken will substantiate the project's claims for the feasibility of the proposed WARRANT system. He can be reached at the Department of Computer Science, Carnegie-Mellon University, Pittsburgh 15213, (412) 578-2984.

Jill Larkin, Associate Professor of Cognitive Psychology and Director of Carnegie-Mellon's Center for the Design of Educational Computing, will testify to the educational significance of the project and the resources available for its critical scrutiny and support in this environment. She can also provide perspective on the dissemination prospects for the project via the Inter-University Consortium for Educational Computing and the recently founded CMU/McGraw-Hill series for innovative educational software. She can be reached at (412) 578-3785.

John P. Crecine, Senior Vice President for Academic Affairs and Director of Computing and Planning for the university, can testify to the institutional commitment to continued excellence in and support of educational computing. He can also substantiate the commitment, competence, and enterprise of the project principals in the area of educational computing and innovation. He can be reached at (412) 578-2056.

Carnegie-Mellon affirms its strong support for this project with a cost-sharing contribution that comes to 44% of the total costs. (See the budget section).
2.6. Discussion of Plans for Wider Impact

Carnegie-Mellon University, perhaps more than any other educational institution in the world, has made strong and lasting commitments to the development and dissemination of educational software. Under a Carnegie Corporation grant, Carnegie-Mellon founded an inter-university consortium for educational computing (ICEC). It recently signed a contract with IBM to develop, collaboratively, a new generation, low-cost personal computer as well as to develop a campus-wide network that would allow students owning their own computer to be able to "talk to" virtually anyone, virtually anywhere, on campus. The responsibilities of these development and networking activities fall to the university's recently constituted Information Technology Center (ITC). (See appendix III for details on this and the other organizations we reference later in this section). Jim Gosling, a systems designer for the ITC, will serve as a technical consultant for our project.

Carnegie-Mellon has also established an internal organization to supervise and aid in the development of educational software—the Center for Educational Computing (CDEC). Headed by Jill Larkin, a cognitive psychologist and partly funded by a Sloan grant to integrate technology and liberal learning, CDEC is presently negotiating with Ruth Von Blum (a major consultant for our project) to assume the position of associate director. Our project has been carefully reviewed and supported both by members of the ITC and CDEC. In addition, Carnegie-Mellon recently signed a long-term agreement with McGraw-Hill to establish a series of books on educational software developed at Carnegie-Mellon. We have been in close contact with editors of McGraw-Hill who are not only interested in publishing the proposed Covey, Kaufer text but who are also interested in publishing a proposed book by Neuwirth and Kaufer on the general design and implementation of "learning environments" for writers.

The task of writing the documentation that will accompany the WARRANT system (as well as all the educational software to come out of Carnegie-Mellon) falls on the Communication Design Center (CDC). Headed by John R. Hayes, a cognitive psychologist who will serve as a consultant on the empirical aspects of our project, the CDC has gained a national reputation for writing user-tested (with protocols), user-friendly, easy-to-read documentation. Much of the work we undertake with WARRANT will be publicized as part of the activity of the university's Writing Center. The Writing Center is funded internally and, in part, by a grant from the Buhl foundation. The chief function of the Writer Center is to evaluate and develop
improved software for reading and writing.

All of the organizations we have mentioned (ICEC, ITC, CDEC, CDC, and the Writing Center) testify to the institution's enthusiasm for seeing the WARRANT system make a significant impact on educational computing in the humanities.

In many books, Walter Ong has pointed out the discomforts and displacements that arise when new technologies are introduced into education. Living in an oral culture, Plato thought writing would destroy a student's memory (and in some sense, it has). In cultures where print is a luxury or taboo, books have been thought to destroy the moral fiber of the young. Each technological innovation has had its pluses and minuses, but people adapt and ultimately learn how to adjust. We are living in a time when computers are doing more and more of the world's serious work, but also a time when they remain novelties for education.

Like all strange technologies that wend their way into education, we have either overblown expectations about them or irrational hatred. We think they either must solve all our problems or create them. We haven't yet learned to adjust. Our fondest hope for this project is that it eases these adjustment pains. Computers can't solve the literacy crisis but, used judiciously, they can help us learn more about how to solve it than we could learn without them. That is why we have designed the WARRANT system to be as much a learning tool for teachers of reading, reasoning and writing as it is for students.
3. Budget Summary Sheets

[See pages that follow]
4. Budget Detail and Narrative

4.1. DIRECT COSTS

4.1.1. Salaries and Wages

YEAR 1 (January - December 1985)

Preston Covey
1/4 release @AY '84-'85 base
1/9 summer @AY '85-'86 base

David Kaufer
1/4 release @AY '84-'85 base

Christine Neuwirth
1/4 release @AY '84-'85 base

Systems Programmer @CY Full

Secretary @CY 3/4-time

Sub-totals

FIPSE

CMU

69,708

3,000

YEAR 2 (January - December 1986)

Preston Covey
1/2 release @AY '85-'86 base
1/9 summer @AY '86-'87 base

David Kaufer
1/2 release @AY '85-'86 base
1/9 summer @AY '86-'87 base

Christine Neuwirth
1/2 release @AY '85-'86 base
1/9 summer @AY '86-'87 base

Cheryl Geisler @ 1/2-time

Systems Programmer @CY Full

Secretary @CY 3/4-time

Sub-totals

FIPSE

CMU

91,739

26,157

YEAR 3 (January - December 1987)
Preston Covey  
1/2 release @AY '86-'87 base  
2/9 summer @AY '87-'88 base  
1/4 release @AY '87-'88 base  

David Kaufer  
1/2 release @AY '86-'87 base  
1/9 summer @AY '87-'88 base  
1/4 release @AY '87-'88 base  

Christine Neuwirth  
1/2 release @AY '86-'87 base  
1/9 summer @AY '87-'88 base  
1/4 release @AY '87-'88 base  

Cheryl Geisler @ 1/2-time  

7,260  

Systems Programmer @CY Full  

48,400  

Secretary @CY 3/4-time  

7,260  

3,630  

Sub-totals  

107,710  

60,143  

TOTALS for Salaries & Wages  

269,157  

89,300  

4.1.1.1. Salary and Wages Narrative  
Release time at 1/4 the AY 1984-85 salary base of the PI's provides half-time release for the first (spring) semester of the project. Release time at 1/2 the AY's '85-'86 and '86-'87 provides half-time release each of those years, presumably to be concentrated in the spring semester. Release time at 1/4 AY '87-'88 provides half-time release during the final (fall) semester of the project.  

The university provides half the release time requested for the project, picking up the final semester while FIPSE is asked to cover the first semester of the project. AY '84-'85 salary bases are: Increments on academic line salaries are figured at 15%, where the range of raise for the principals has been 10% - 20% in the past. Increments on other salaries and compensation are figured at 10%, where salaries are not functions of productivity or merit raises do not apply.  

We fully expect that the project PI's will spend two or more summer months on the project each summer. Project Director Covey will have responsibility for over-all project coordination, budget and accounting, editing and quality-control on all project products and documentation, and collaborative relations with the various
administrative and support units of the university and the Inter-University Consortium for Educational Computing (ICEC — see Appendix III), as well as for his share of the substantive, research, and curriculum-development dimensions of the project. An additional summer month is requested for Covey during the first and last summers of the project. Summer support is calculated at 1/9 the salary of the subsequent academic year for each summer month.

For the first year of the project, Geisler, our Senior Research Assistant (responsible for running our protocol research subjects), will be ABD and supported as a graduate student (at the top rate for advanced research assistants in the college @ 6,600 for the academic year and 2,000 for the summer). After receiving her Ph.D. in January of 1986, she will be employed at a half-time rate for each of the succeeding years of the project.

One-third of the requested secretarial support will be provided by the university. It is expected that one-quarter of a full-time secretary will be required to assist with the administration of the project; half a full-time secretary will be dedicated to transcribing protocols and typing other project documentation. We require the use of secretaries who are familiar with the text-processing software on both our DEC 20 and VAX mainframes and the IBM Personal Computer, file management and transfer, etc. We expect to split the clerical duties between secretaries attached to Covey and the Communication Design Center, respectively.

Absolutely crucial to our project, the feasibility and speed with which we can implement a prototype of WARRANT for formative evaluation and class use, is a top-flight, full-time, dedicated systems programmer. In particular, we need a programmer familiar with the sophisticated information- and frame-management program (ZOG) that will be built into WARRANT, and who can quickly master the advanced software under development for the IBM Peach workstation on which we will develop WARRANT. For this reason, we want to draw on local talent, a programmer who has extensive experience with the requisite local software. Our consultants in Computer Science, the ITC (see Appendix III), and the ZOG Project working group estimate that even an extremely talented programmer would require a year to begin to master the ZOG system alone. (This was the case for Christine Neuwirth.)

The high cost of a Research-Scientist class programmer will be offset by precious time saved, the guarantee of a prototype WARRANT system well within the first year
of the project, and a quality system by the project’s third year. For this or any very ambitious, standard-setting computing project, the importance of expeditious program implementation and a first-class programmer cannot be overemphasized. We need fast and reliable turn-around on implementing our system design in order to make rapid progress in the development and assessment of our curriculum. Far less ambitious projects might make do with a lesser programmer, but this is one item we cannot stint on. We have effectively traded off release time that we could surely use for the sake of expediting the implementation of the system. We defined the systems programmer budget line by the rate of the senior research scientist we want to enlist from Mellon Institute of Science. This is a job for a computer scientist, not a clever hacker.

4.1.2. Employee Benefits

<table>
<thead>
<tr>
<th></th>
<th>FIPSE</th>
<th>CMU</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>YEAR 1</strong> (January - December 1985)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Benefits @ 24%</td>
<td>16,690</td>
<td>720</td>
</tr>
<tr>
<td><strong>YEAR 2</strong> (January - December 1986)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Benefits @ 26%</td>
<td>22,136</td>
<td>6,800</td>
</tr>
<tr>
<td>@ 10% (part-time)</td>
<td>660</td>
<td>660</td>
</tr>
<tr>
<td>Sub-total</td>
<td>22,796</td>
<td></td>
</tr>
<tr>
<td><strong>YEAR 3</strong> (January - December 1987)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Benefits @ 28%</td>
<td>28,126</td>
<td>16,840</td>
</tr>
<tr>
<td>@ 10% (part-time)</td>
<td>726</td>
<td>726</td>
</tr>
<tr>
<td>Sub-total</td>
<td>28,852</td>
<td></td>
</tr>
<tr>
<td>TOTAL Benefits</td>
<td>68,338</td>
<td>24,360</td>
</tr>
</tbody>
</table>

4.1.2.1. Benefits Narrative

The ten percent benefits rate in the second and third years applies to part-time personnel, in this case Geisler, who will stay on after completing her Ph.D. to continue to serve as our Senior Research Assistant.
4.1.3. Travel

YEARS 1 (January - December 1985)

Round trip Pgh.-D.C. + Hostelry for one Project Director 480

Round-trip air L.A.-Pgh. for Ruth Von Blum 600

Sub-total 1,080

YEARS 2 (January - December 1986)

Round trip Pgh.-D.C. + Hostelry for one Project Director 530

Round-trip air L.A.-Pgh. for Ruth Von Blum 660

Sub-total 1,190

YEARS 3 (January - December 1987)

Round trip Pgh.-D.C. + Hostelry for one Project Director 580

Round-trip air L.A.-Pgh. for Ruth Von Blum 726

Sub-total 1,306

TOTAL Travel 3,576

4.1.3.1. Travel Narrative

For the FIPSE Project Directors' conference in D.C.: the first year, we assume 150.00 for round-trip airfare, 75.00 per diem for hotel, 25.00 per diem for meals, and 30.00 ground transport. Costs for years two and three raised at the rate of 10% per year. For Ruth Von Blum’s consultant trips: we assume that advance reservations will allow savings with special rates and that Ruth will stay with one of the project directors.
4.1.4. Equipment

YEAR 1 (January - December 1985)

One transcribing/tape machine 550
One pair intercom units 110
Four IBM Personal Computers + complement of software
@ 4,000 ea. 16,000
Four Epsom R80 Printers @ 750 ea. 3,000
Five IBM Peach Workstations + resident software
@ 12,000 ea. 60,000

Sub-totals 660 79,000

YEAR 2 [No costs]

YEAR 3 (January - December 1987)

Five IBM Ultimate Workstations + resident software
@ 4,000 ea. 20,000

TOTALS Equipment 660 99,000

4.1.4.1. Equipment Narrative

The tape transcription machine is for recording and transcribing protocols. The latter is a slow, time-intensive activity at best; a machine equipped with a pedal control to start and stop the tape would facilitate the process immeasurably. The intercom units will allow us to monitor a subject's protocol recording without being intrusive; this will obviate the often inhibiting effect of having a researcher present, while allowing the researcher to intrude when and only when the subject is at a loss or stops reporting his thought processes aloud.

The IBM Personal Computers will be used to prepare most of our project documents (thus saving computing costs on the mainframe), and to review other writing/reasoning software. Assessment of extant software is an important part of
our project. The college will provide these machines to encourage migration of demand from the increasingly expensive mainframe systems.

The five IBM Peach workstations (one for each PI, the programmer, the research assistants and subjects) will be the development vehicles for the WARRANT system, which will be ported in the third year of the project to the ultimate IBM workstation planned for campus-wide distribution beginning in 1986. We are not assuming that the ultimate workstation will actually be available for student use by 1986; but our project group will be allocated a complement of these machines when they become available for development work, whether that be '86 or '87. We have conservatively supposed it will be the latter. The sooner they are available, the more expensive they are apt to be. The university estimates that these machines will eventually be obtainable for about 3000, but 4000 is a realistic estimate for the initial units. The intermediate, prototype Peach machines, however, are estimated to cost between 10,000 and 15,000. The ITC (see Appendix III) thinks 12,000 a fair estimate for purposes of cost-out. The facilities and capacities of the Peach workstation are described in Appendix II. The ITC and the college will share the cost of installation; the ITC will bear the cost of maintenance in return for the field testing and feedback they will receive in our development work (see "Other," Section 4.1.7 below).

4.1.5. Materials and Supplies

<table>
<thead>
<tr>
<th></th>
<th>FIPSE</th>
<th>CMU</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>YEAR 1 (January - December 1985)</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Software/peripherals</td>
<td>600</td>
<td></td>
</tr>
<tr>
<td>Technical manuals/textbooks</td>
<td>450</td>
<td></td>
</tr>
<tr>
<td>Postage</td>
<td>180</td>
<td></td>
</tr>
<tr>
<td>Printer paper</td>
<td>100</td>
<td></td>
</tr>
<tr>
<td><strong>Sub-total</strong></td>
<td>1,430</td>
<td></td>
</tr>
<tr>
<td><strong>YEAR 2</strong></td>
<td></td>
<td>1,573</td>
</tr>
<tr>
<td><strong>YEAR 3</strong></td>
<td></td>
<td>1,730</td>
</tr>
<tr>
<td><strong>TOTAL Materials and Supplies</strong></td>
<td></td>
<td>4,733</td>
</tr>
</tbody>
</table>
4.1.5.1. Materials Narrative

We will need to be able to obtain writing software for assessment as it becomes available. Some of this, as at present, we will obtain in exchange for an evaluation (Writing as Thinking was obtained in this manner). The need inevitably arises for special peripherals (circuit boards etc.) and technical manuals. We will also want to be able to review new textbooks as they come out, in the fields of composition, rhetoric, and critical reasoning. Printer paper for work-a-day printing at the IBM PC's should run about four boxes (@ $25 per) a year. (Final formatting and printing of project documents for dissemination will be done on a mainframe and the Xerox laser printer). The cost increment rate for years two and three is ten percent.

4.1.6. Consultants (and Research Assistants)

<table>
<thead>
<tr>
<th>Consultant</th>
<th>FIPSE</th>
<th>CMU</th>
</tr>
</thead>
</table>

**YEAR 1** (January - December 1985)

Ruth Von Blum - 30 Days @ 200 per diem 6,000

John Richard Hayes 6,000

Donald McCracken 6,000

Cheryl Geisler - Senior Research Assistant 8,600

Two graduate student research assistants @ 2,000/semester ea. 8,000

Five Expert Protocol Subjects @ 100 per diem for ten quarter-days ea. 1,250

Five student protocol subjects per semester @ 30 hours ea. @ 3.50 per hour 1,050

Sub-totals 24,900 *

**YEAR 2** (January - December 1986)

Ruth Von Blum - 30 Days @ 200 per diem 6,000
<table>
<thead>
<tr>
<th>Consultants/Assistants</th>
<th>Year 3 (January - December 1987)</th>
<th>Sub-total</th>
</tr>
</thead>
<tbody>
<tr>
<td>John Richard Hayes</td>
<td></td>
<td>6,000</td>
</tr>
<tr>
<td>Donald McCracken</td>
<td></td>
<td>8,800</td>
</tr>
<tr>
<td>Two graduate student research assistants @ 2,200/semester ea.</td>
<td></td>
<td>8,800</td>
</tr>
<tr>
<td>Five Expert Judges @ 100 per diem for six days ea.</td>
<td></td>
<td>3,000</td>
</tr>
<tr>
<td>Sub-total</td>
<td></td>
<td>17,800</td>
</tr>
<tr>
<td></td>
<td>Year 3</td>
<td>62,550</td>
</tr>
</tbody>
</table>

**TOTAL Consultants/Assistants:** 62,550
4.1.6.1. Consultant (and Research Assistant) Narrative

* The asterisked lines represent the uncharged, *pro bono* contributions of our in-house consultants who have a professional research interest in our project: Hayes, in the protocol research and user documentation design and evaluation; McCracken, in our adaptation of the ZOG information- and frame-management program to the WARRANT environment.

Ruth Von Blum is our principal external consultant. She will spend at least a month a year, two or more weeks on annual site visits and the rest writing curriculum and evaluation, reviewing other software and projects. She will be a principal reviewer of the "user friendliness" of our system and a system and curriculum design consultant. We will hold "story board" sessions during her visits (see Appendix II on our use of the "story boarding" process in system and curriculum design). We include other non-salaried personnel under this category—our Senior Research Assistant (during her final year as a graduate student), the two graduate research assistants, expert subjects and judges, and novice/student subjects, whose roles are outlined in the workplan of the project (Section 2 above).

### 4.1.7. Other

<table>
<thead>
<tr>
<th>FIPSE</th>
<th>CMU</th>
</tr>
</thead>
<tbody>
<tr>
<td>YEAR 1 (January - December 1985)</td>
<td></td>
</tr>
<tr>
<td>Machine maintenance</td>
<td>4,200</td>
</tr>
<tr>
<td>Printing (computer/copier)</td>
<td>1,200</td>
</tr>
<tr>
<td>Telephone</td>
<td>600</td>
</tr>
<tr>
<td><strong>Sub-total</strong></td>
<td><strong>1,800</strong></td>
</tr>
</tbody>
</table>

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>YEAR 2 (January - December 1986)</td>
<td></td>
</tr>
<tr>
<td>Machine maintenance</td>
<td>4,200</td>
</tr>
<tr>
<td>Printing (computer/copier)</td>
<td>1,800</td>
</tr>
<tr>
<td>Telephone</td>
<td>600</td>
</tr>
<tr>
<td><strong>Sub-total</strong></td>
<td><strong>2,400</strong></td>
</tr>
</tbody>
</table>

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>YEAR 3 (January - December 1987)</td>
<td></td>
</tr>
</tbody>
</table>
Machine maintenance 5,600
Printing (computer/copier) 2,400
Telephone 600
Sub-total 3,000

TOTALS Other 7,200 14,000

4.1.7.1. Narrative

The maintenance of the IBM intermediate and ultimate workstations will be the responsibility of the ITC (see Appendix III). Especially in their formative development stage, it is anticipated that the machines will require trouble-shooting and maintenance. The cost for this item is figured on the rate allowed for repair and maintenance of project apparatus and machinery by the NSF: 14 percent of the capital equipment costs. We have halved this rate. This gives us a cost per unit per year that is half-again the internal maintenance contract rate for the less sophisticated IBM XT. (Among the differences, the IBM Peach has two fixed disks vs. the XT’s one; these are among the components most susceptible to malfunction and quite expensive to replace or repair.)

The copying will be of relevant research literature and project documents (reports, manuals, subject instructions and protocols, etc.). Printing of successive drafts of our textbook material and other materials or publications for dissemination will be done on the Computation Center’s mainframe and laser printer. The volume will be considerable and will increase over the course of the project.

4.1.8. Total Direct Costs

<table>
<thead>
<tr>
<th>Item</th>
<th>FIPSE</th>
<th>CMU</th>
</tr>
</thead>
<tbody>
<tr>
<td>Salaries and Wages</td>
<td>269,157</td>
<td>89,300</td>
</tr>
<tr>
<td>Benefits</td>
<td>68,338</td>
<td>24,360</td>
</tr>
<tr>
<td>Travel</td>
<td>3,576</td>
<td></td>
</tr>
<tr>
<td>Equipment</td>
<td>660</td>
<td>99,000</td>
</tr>
<tr>
<td>Materials/Supplies</td>
<td>4,733</td>
<td></td>
</tr>
<tr>
<td>Consultants/Assistants</td>
<td>62,550</td>
<td>14,000</td>
</tr>
<tr>
<td>Other</td>
<td>7,200</td>
<td></td>
</tr>
<tr>
<td><strong>TOTALS</strong></td>
<td><strong>416,214</strong></td>
<td><strong>226,660</strong></td>
</tr>
</tbody>
</table>
4.1.9. Modified Total Direct Costs (MTDC)

<table>
<thead>
<tr>
<th></th>
<th>FIPSE</th>
<th>CMU</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total Direct Costs</td>
<td>416,214</td>
<td>226,660</td>
</tr>
<tr>
<td>Less capital equipment costs over 500 per unit</td>
<td>- 550</td>
<td>-99,000</td>
</tr>
<tr>
<td>Less 50% graduate student compensation</td>
<td>-17,100</td>
<td></td>
</tr>
<tr>
<td>Modified Total Direct Costs</td>
<td>398,564</td>
<td>127,660</td>
</tr>
</tbody>
</table>

4.2. INDIRECT COSTS

<table>
<thead>
<tr>
<th></th>
<th>FIPSE</th>
<th>CMU</th>
</tr>
</thead>
<tbody>
<tr>
<td>25% on FIPSE MTDC</td>
<td>99,641</td>
<td></td>
</tr>
<tr>
<td>30% on FIPSE MTDC</td>
<td>119,569</td>
<td></td>
</tr>
<tr>
<td>55% on CMU MTDC</td>
<td>70,213</td>
<td></td>
</tr>
<tr>
<td>TOTALS Indirect Costs</td>
<td>99,641</td>
<td>189,782</td>
</tr>
</tbody>
</table>

4.3. TOTAL PROJECT COSTS

<table>
<thead>
<tr>
<th></th>
<th>FIPSE</th>
<th>CMU</th>
</tr>
</thead>
<tbody>
<tr>
<td>Direct Costs</td>
<td>416,214</td>
<td>226,660</td>
</tr>
<tr>
<td>Indirect Costs</td>
<td>99,641</td>
<td>189,782</td>
</tr>
<tr>
<td>TOTALS Project Costs</td>
<td>515,855</td>
<td>416,442</td>
</tr>
</tbody>
</table>

PROJECT TOTAL = 932,297

[The discrepancies of one dollar between the totals in the budget detail and the three annual budget summaries (reflected on the title page) are the result of arriving]
at these figures by different sequences of multiplication and division and rounding off to the nearest whole dollar amount above or below $.50.]
5. APPENDIX I.

WARRANT: Pedagogy and Curriculum Testing

A significant part of our proposed project is to elaborate an effective pedagogy for integrating the teaching of critical reading, reasoning, and writing skills, to research the cognitive processes presupposed and addressed by this pedagogy, and to implement this pedagogy in the dynamic computer environment of WARRANT as well as in the textbook that Covey and Kaufer will write for this project.

Much more conceptual and empirical work of just the sort we propose is needed on how best to define and teach the complementary and interactive skills of critical reading, reasoning, and writing. What follows is only an outline of the pedagogical strategy and educational principles we propose to follow, elaborate, and implement; a prototype pedagogy, with some provisional, speculative illustration of the type of curriculum we will seek to refine and develop for WARRANT.

Needless to say, we cannot adequately illustrate in linear, discursive prose the dynamics of a computer environment we have yet to implement. In particular, it is impossible to represent the facilities by which complex and hierarchical options can be simultaneously represented and accessed with the dynamic windowing capacities resident in WARRANT. Rather, we attempt here to sketch the pedagogy that will inform our efforts, with some modest illustration of the kind of curriculum, exploration, and options it will allow. We also rehearse the programme of learner-centered and expert-model protocol research that will both inform the development of our pedagogy and provide critical feedback for the formative evaluation of the WARRANT system and its pedagogy.

5.1. The Educational Theory that Informs the WARRANT System

5.1.1. A Process Model of Skill Acquisition

Students need to learn a great many things to become competent at any skilled, goal-directed task—including the goals of the task, strategies to meet these goals, and tests to determine whether these strategies have met with success or failure.\(^5\)
5.1.1.1. Goals

Students must know what goals to select during writing, reading, and reasoning. Several researchers have examined how the failure to set writing goals leads to writing problems. Linda Flower and John R. Hayes, for example, have found that inexperienced adult writers, unlike their experienced counterparts, often fail to set goals to consider their audience.6 Witte and Faigley have speculated that college students fail to produce coherent texts because they fail to set goals to elaborate sufficiently on what they want to say.7 And Scardamalia and Bereiter, in a similar vein, have suggested that writers may lack a “repertoire of potentially attainable goals to which a particular task may be assimilated.”8

5.1.1.2. Strategies

Students must know how to pursue their goals. The emphasis in the New Rhetoric on developing specific heuristics to aid students’ writing processes has come as educators and researchers acknowledge the importance of moving beyond vague goal-directives such as “Be clear” to explicit, operational suggestions.9

5.1.1.3. Discriminations

Students must know when they have succeeded or failed to meet their goals. Empirical evidence strongly suggests that students often fail to recognize whether the text they produce succeeds or fails to meet their writing goals. Perl and Bartholomae independently report that unskilled writers typically fail to perceive mistakes in their own text.10 In recent work on revision, Hayes has found that unskilled adult writers often overlook problems in texts they are asked to revise.11

5.1.1.4. Scheduling goals

Yet the most difficult obstacle to learners of reading, reasoning, and writing may well be the learning of effective schedules for setting goals. The scheduling problem has been commented on extensively in the literature on the writing process. In their cognitive process model of writing, Flower and Hayes have described the writing process as recursive, interruptable, and modifiable, and detailed the difficulties writers experience managing the scheduling of goals.12 Scardamalia and Bereiter have noted the difficulty children face in integrating their multiple intentions for a text.13 R.J. Bracewell has conducted several studies that show how writers’ knowledge about writing often exceeds their performance in an actual writing tasks.14 Unlike simpler skills that can be learned in a linear schedule with a rigid agenda of independent goals (e.g., tying one’s shoe, adding numbers, spelling words) reading, reasoning, and writing involve many overlapping tasks. These tasks, in turn, involve many
overlapping subtasks. And these subtasks, in turn, involve many overlapping goals and strategies. Managing their scheduling can be overwhelming.

The pedagogy and computer environment we are proposing will address as crucial each of the skills we have identified in the acquisition of reading, writing, and reasoning skill.

5.1.2. Learning by Doing and Discovery Learning

Teachers know that students learn much, if not most, of their skill from experience. The role of experience in acquiring math and science skill is widely acknowledged by cognitive psychologists,15 and Dewey's argument for the role of experience in all types of education is familiar.16

5.1.2.1. Structured Learning

Teachers also know that students typically require a structure to help them to learn on their own, to help them understand the various options available to them, to help them work through and evaluate the suitability of these options, and to help them "abort" options that don't seem to be helping as well as "return" to those that do. Some teachers, however, thinking that structure must be imposed at the expense of freedom, fail to distinguish "giving students ways of structuring different schedules" from simply "giving students a schedule."17 Instead of providing students with "ways to go about it—proceed as you will," they provide students with the more linear directive "here's how to do it." This approach stifles self-discovery and learning by doing. On the other hand, the more romantic tradition in education that merely offers "proceed as you will" fails to give students the boost they need to begin to learn on their own. Writing practioners have not been immune to these problems.18

5.1.2.2. Freedom with Structure

We believe that both freedom and structure are important, and our philosophy of education reflects this belief. Students learn by discovering what they are doing in the act of doing it. They need to be able to experiment with their own schedules, to trouble-shoot their own problems, to learn from their mistakes, and to adopt strategies suited to their own strengths, weaknesses, and individual styles. Having to reconcile this mild tension between freedom and structure underlies in part why teachers find processes of reading, writing, and reasoning so difficult to teach.

We think that the following guidelines explain how this tension can be reconciled: Students should be provided with only as much structure as they require to stimulate
productive use of their own processes. Teachers, not to mention developers of educational software, need to be sensitive to the tradeoffs between freedom and external control in facilitating writing, reading, and reasoning skill. The design of WARRANT will be informed by these guidelines.

5.2. The Educational Theory that WARRANT Can Inform

The most exciting prospect of the WARRANT system is that it can help us extend our understanding of the approaches that are optimal in teaching reading, reasoning, and writing. As educators, we have much more to learn from the WARRANT system about how students acquire skills of reading, reasoning, and writing, than we presently know.

5.2.0.1. The Traditional Problem of Curriculum Testing

Since philosophers and rhetoricians first decided to teach their art centuries ago, pedagogies have been formulated to teach critical thought and expression—pedagogies that survive to a greater or lesser degree in our college textbooks. What characterizes these pedagogies, moreover, is that they represent logical reconstructions of what the effective thinker or writer must know, not real-time advice or feedback about what the effective thinker or writer should be doing at this moment or that. For example, the reader of Aristotle’s *Ethics* learns about the types of knowledge that informs ethical skill—but not how to apply this skill on a real-time ethical problem; the reader of Aristotle’s *Rhetoric* learns about the types of knowledge that informs rhetorical skill—but not how to apply this skill on a real-time rhetorical problem.

Teachers have always been aware of the distinction between instruction based on rational reconstruction and instruction based on real-time advice or feedback. The first type of instruction is the stuff of curriculums and textbooks. The second type of instruction has been traditionally thought to depend on the individual skill, experience, and sensitivity of a specific teacher.

In the best of all possible worlds, these types of instruction are not only complementary but mutually reinforcing. A good curriculum or textbook should be consistent with and inspire good interactive instruction. Interactive instruction, on the other hand, would seem to be made only more effective when put in the service of a worthwhile curriculum or text. Unfortunately, these are only ideals in the practical world of education—and untestable ideals at that.
The fact of the matter is that we have a very poor understanding of the subtle and complex ways in which rational or "noninteractive" instruction affects and is in turn affected by interactive guidance. Curriculums, textbooks, and even teachers standing before a class can only spray words here and there. Students, embarking on a real assignment, need to make hundreds of decisions. How do the "rational" words of the educator influence those student decisions, if at all? If there is an influence, is it for the good or not? These are the questions that need to be addressed before we can start assessing the effects of curriculum and texts on student learning.

5.2.0.2. WARRANT as an Environment to Test Curricula

With the advent of new technologies to facilitate writing, reading, and reasoning, answers to these questions are in the offing. An interactive technology forces us to "operationalize" a curriculum, to take it out of the clouds of "useful knowledge" and to see what students actually do with it, and the technology itself can help us observe them. So much the good for technology. But if the technology is itself too rigid, we can't make fair assessments of the instructional techniques it attempts to embody. Thus, technologies can help us test the interactive usefulness of various curricula. Yet unless the technology in question is flexible, it will "force fit" the students' interaction so rigidly that it can never provide a reliable test for that curricula's real worth. We believe WARRANT enjoys all the advantages with far fewer the limitations of previous technologies.

5.3. Curricular Materials to Be Used and Tested on WARRANT

We now consider how we will seed WARRANT'S data base with readings and curricular guides for teaching goals, strategies, and discriminations.

5.3.1. Readings

Once any given topic is chosen that is agreeable to the funding agency and to the principal investigators, several related articles will be selected. The authors of these articles will reflect a variety of perspectives—philosophers, public decision makers, researchers—in order to ensure that the users of WARRANT become familiar with a range of techniques and purposes. Several anthologies on ethical and public policy issues are already available—such as the Gross volume we will use in our illustration below—whose reading selections address common issues from different points of view, interests, and values.
5.3.2. Goals

What goals might a student want or need to pursue in a critical reading, reasoning, and writing task? As long-time instructors in the field, we have made some stabs at answers to this question, and every textbook on the market recommends its own agenda of skills and subskills. Much in these taxonomies is derived from one of two sources: introspection into one's own process, or analysis of what is desired in the final product. We propose to make use of a third source in our project: the "thinking-aloud" protocols of both experts and novices as they work through the task. We believe such protocols will help us to avoid the narrowly simplistic prescription or the rigidly linear agenda.

But we do come to the project with some provisional conceptions of what the task requires and what goals must be pursued to satisfy these requirements. We assume, for example, that all good readers/writers/reasoners dealing with normative prose know to read closely, know to read for a critical summary, know to try to assess strengths and weaknesses in an argument, know to compare and contrast these strengths and weaknesses across articles and so on. More than any other aspect of curricular instruction in reading, writing, and reasoning, the literature is filled with information about what goals students need to set. We shall extract information about what goals students need to be prompted for from the literature as well as our own research.: The following are representative of the array of sources we shall tap:

Some References For Prompting Students for Goals


P. Covey, "Logic and Liberal Learning," Teaching Philosophy, 4:3-4 (July/October, 1981), 207-230.


5.3.3. Strategies/Heuristics

Strategies (sometimes called heuristics) describe the operations students can perform to achieve a goal as well as the useful techniques a student might apply to expedite these operations. Expert strategies in critical reading, reasoning and writing do not guarantee success; they work much of the time, but cannot be applied indiscriminately in all situations.

Very little is known or specifically detailed about the strategies of critical reasoning, reading and writing. Some of the literature from which we will draw information about goals also details strategies. But for the most part, experts develop critical strategies over time as they solve increasingly difficult problems. Their knowledge of these strategies is usually intuitive and difficult to articulate. In the past, educators have had little recourse in teaching strategies other than to give students well-designed assignments that would allow them to "find their own way."

5.3.4. Teaching Discrimination Through Expert and Novice Models

Although the strategies we derive from the thinking-aloud protocols will allow us to give students specific advice about how to proceed, we do not underestimate the difficult time students have applying expert processes directly or knowing whether the application of an expert process has moved them in the right direction. For this reason, students need to learn more than simply "what experts do." They also need to learn how to apply this knowledge effectively by learning how to discriminate productive and nonproductive uses of the expert strategy. How can students learn how to make these discriminations? Through models. We will give students edited versions of selections from the protocols of both experts and novices as they pursue particular goals and follow selected strategies; Without necessarily identifying for students which models are the experts and which are the novices, we will give them practice distinguishing productive applications from nonproductive ones.
These models make available several types of information that we do not think can be had elsewhere. First, they show how a strategy is worked out in a particular context and is adapted to a particular situation—they give students important discrimination training. Second, they disabuse students of the notion that strategies are infallible rules rather than flexible rules-of-thumb—they teach students about the problem-solving character of their task. And third, they help students see how to adapt strategies to their own strengths and limitations—they not only discover how those with years of experience proceed, but they can also study the processes of those with knowledge and concerns closer to their own—their peers.

5.3.5. Teaching Discrimination Through Assessments of Models

To facilitate discrimination training even further, we believe it useful to give students assessments of expert and novice models along with the models themselves. Therefore, we will also provide students access to edited transcriptions of discussions by a panel of experts on each of the models. These transcriptions will provide students with the feedback that they need in order to test their own discriminations and understandings of alternative goals and strategies.

5.4. The Research to Warrant WARRANT

5.4.1. Methodology

"Thinking-aloud" protocols will be our primary source for collecting information about strategies, models, and assessments of how well (or poorly) these models fared.

5.4.2. Our Selection of Readings

While our illustrations below discuss the topic of reverse discrimination and draw from the anthology edited by Barry R. Gross, we are not committed to this topic for the final project. However, since we are restricting our focus to normative argumentation, any issues we use must meet the following criteria: They must allow for two defensible positions, involve a conflict of principles rather than words or facts, and they must invoke a realm of experience that a college freshman can easily understand and relate to. The issue of reverse discrimination passes all three tests; but other normative issues do as well, and we are open to suggestion.
5.4.3. Our Selection of Models

We plan to ask two types of people to do a critical reading, reasoning, and writing task for us, a set of 5 experts and a set of 5 novices. Experts will be drawn from those faculty at Carnegie-Mellon and environs who are acknowledged to be expert thinkers and teachers in fields of critical reasoning. Novices will be drawn from a population of students who are taking either the introductory philosophy course or introductory writing course at Carnegie Mellon. They will be selected for their interest and their lack of experience in the task.

5.4.4. Phases of Our Protocol Research

5.4.4.1. The First Phase

In the first phase of research, we will ask our experts to work closely with us as they work through selected critical reading, reasoning and writing tasks. At each "thinking-aloud" session, they will be free to pursue their own goals and strategies in completing the task. After they have completed each session, we will ask them for retrospective protocols in which they talk about their own protocols. After all have finished this assignment, we will have them meet together as a group to discuss the task and strategies associated with it.

We will segment and classify each of these expert protocols according to the goals they pursued and the strategies they used. We will then attempt to represent the problem space of the assignment and try to characterize the various paths that our subjects took within that space.

5.4.4.2. The Second Phase

In the second phase, we will ask our novices to work through the same assignment. Like the experts, we will initially let the novices work on their own with little intervention. Unlike the experts, however, if we find them making little headway, we will be quick to intervene and tell them what sub-task to work on (based on the taxonomy developed in phase 1) in each session. They will also be given some initial tutoring in possible strategies to eliminate gross misunderstandings. We will then ask them for retrospective protocols and ask them to meet as a group to discuss common and divergent strategies and advice they might give others doing the assignment.
5.4.4.3. The Third Phase

In the third phase of data-gathering, we will ask our experts to meet as a panel to review, discuss, and assess the protocols of both experts and novices. We will ask them to comment on such things as the productivity of the processes, the pertinence of the goals they see being pursued by each writer, and the factors that affect the outcome of each attempt. These assessments will be distilled and made available as part of the structured data base of WARRANT.

5.5. A Hypothetical Interaction Session

To illustrate how WARRANT is designed to facilitate learning with these curricular materials at hand, it will be useful to imagine how a particular student might interact with it.

5.5.1. Some Qualifications and Caveats

We need to make three important qualifications about the following discussion. First, the order of activities we describe is intended to suit only one student—other students will have different patterns of interaction and each student’s interaction will be unique. Second, our illustrations of what the student might see on the computer screen are provisional and speculative. It is virtually impossible to reproduce here the high resolution displays and multiply windowed tiles that students will actually interact with. Indeed, the reader will see menu systems in the following figures that seem to occupy the entire screen. In fact, these menu displays will take up only small portions of the screen. Furthermore, many windows will be able to “open up” on the screen simultaneously, each giving the student a variety of menu options to choose among.

Moreover, any curriculum that we encode in WARRANT will be as much under scrutiny as any other aspect of the system software or hardware. Part of the reason we plan to use protocols of students and experts working through the system is to get a firmer empirical understanding of what curricular materials might prove most effective in interaction with students. In short, the reader should take the following illustrations as extremely tentative and provisional. We are not satisfied with the completeness of our illustration nor do we expect the reader to be. But then, again, were we now able, to our full satisfaction, to draw upon illustrations of how the system will work, we would have had less reason to draw up this proposal.

With these qualifications in mind, let’s follow one student, Janice, as she uses
WARRANT to start a critical reading, reasoning, and writing assignment.

The first time Janice enters WARRANT, we assume she will have only a vague understanding of what she has to do (say, "write a critical paper on the topic of reverse discrimination"). She is not sure what goals she must accomplish, how to accomplish them, or what constitutes a "good" result. The first frame that will appear on her computer screen, therefore, breaks the assignment into its requisite parts, explains what kind of functions WARRANT will perform, and offers a number of entry points into the assignment. (See Figure 1)

These two entry points represent the ways we think Janice might want to start her assignment. She may wish to start reading one of the texts on which the assignment is based, or she may wish to see an overview of the all tasks she needs to accomplish.

Let's assume Janice decides to plow right into the reading to see what the topic of reverse discrimination is about. She would then receive a menu of reading selections to choose from (See Figure 2). Assume she selects the article by Roger A. Shiner on "Individuals, Groups, and Inverse Discrimination." The first few paragraphs of the article appear on her screen (See Figure 3).

As Janice is reading these paragraphs, she may choose to see what options are available to her by asking for a "help" menu which appears at the bottom of her screen (See Figure 4). The menu tells her she may make a note to look up a problematic word in a dictionary, look at the Nickel or Cowan articles cited by the text, examine how a model read through the same paragraphs, see a list of goals that previous readers have pursued in this section of the text, choose another text to read, or change to another task altogether.

If Janice chose to see the Nickel article, it would appear in the bottom half of her screen and she could read through as much of it as she wanted. If she decided that she would rather focus all of her attention on it, she could drop the Shiner article from the top half of her screen. Then all of the options relevant to the Nickel article would be available to her just as if she had chosen it from the menu of readings (Figure 2) to start with. It will be easy for Janice to flip from article to article since all the articles will be indexed on a comparison for key points.

Let's assume that Janice wants to see a list of goals that other writers have
A CRITICAL READING, REASONING, AND WRITING ASSIGNMENT

Your assignment will be to write a critical paper on the topic of reverse discrimination. This means reading several article about the topic, reasoning critically about these articles, and then writing your own paper evaluating them.

WARRANT, the writing environment you are working in, will help you in several ways. First, it will be a place where you can take your notes, make your scribbles, and write your paper.

Second, it will help you to keep track of these notes and organize them as you would like them.

Thirdly, WARRANT will help you to read, to reason and to write by providing you with information about the following:

- TASKS you have to accomplish in order to finish the assignment.
- POSSIBLE GOALS which may help you to accomplish your tasks
- EXPLANATIONS of what these goals are
- STRATEGIES for accomplishing the goals
- MODELS of how other writers (both students and teachers) accomplished their goals
- ASSESSMENTS on how well each of these models or products fulfilled the writer's goals

It will be up to you to decide what kind of information you want at any particular time.

You may enter the assignment in one of two ways:

- through an article to read
- through a task you want to accomplish

Select the type of entry point you want:

Figure 5-1: Janice's first encounter with WARRANT pursued using these paragraphs. For the first few paragraphs of the Shiner article, WARRANT would offer a list of goals aimed at understanding the author's opening moves (See Figure 5)

This menu of possible goals represents our instantiation of the "ways to go about it—proceed as your will" philosophy outlined above. But Janice doesn't understand
These are the articles that you must evaluate for your assignment.

- *Discrimination in Higher Education: A Debate on Faculty Employment*, Miro M. Todorovich and Howard A. Glickstein.
- *Discrimination and Morally Relevant Characteristics*, James W. Nickel.
- *Individuals, Groups, and Inverse Discrimination*, Roger A. Shiner

Select the option you want:

---

Figure 5-2: The menu of reading selections on reverse discrimination.

this task or the set of possible goals and so cannot make a choice between them without help. So she chooses to receive an explanation (See Figure 6).

The explanation of an author's opening moves gives Janice an overview of the intentions of each of the goals. The descriptions are written to enable her to compare the goals with each other and choose between them. Any and all of these goals might be appropriate for the Shiner paragraphs; she can either select one to pursue further or drop her inquiry into relevant goals and return to read more of the text.

The point here is that, with this explanation, Janice has a basis to make a choice. Goal menus combined with explanations are WARRANT's way of giving students information about the first component of skill: the selection of goals.

Information about the second component of skill, the execution of strategies to fulfill goals, is delivered in the form of specific heuristics tied to each of the goals. Once Janice has read the explanation of the goals relevant to understanding an author's opening moves, she can return to the goals menu (Figure 6) and select a goal to pursue. Janice chooses to "identify the generally-held position," and receives a list of pertinent strategies (See Figure 7).

These strategies are heuristics for reading. If Janice calls the Shiner text back into
Roger A. Shiner

INDIVIDUAL, GROUPS AND INVERSE DISCRIMINATION

Many morally sensitive people find themselves faced with the following dilemma. On the one hand, they are persuaded by the argument that if being black, for example, is morally irrelevant, then it is morally irrelevant and no more justifies favorable inverse discrimination than it justifies unfavorable discrimination. On the other hand, this move seems to open the way to neglect, whether benign or malign, of genuine social injustices. James W. Nickel and J. L. Cowan have done much to bring the logic of this situation to the surface. I shall not resist their general strategy of showing that the above is a false dilemma. My concern is with Cowan's diagnosis of the trouble as consisting in the illegitimacy of the thought that blacks as a group deserve inverse discrimination. His view is that one cannot argue that blacks as a group deserve retribution without also implying that blackness as such is a morally relevant characteristic. This is false.

In the first place, it simply is not true that sense can never be made of the thought that a group as such deserves inverse discrimination. Consider these cases. (1) The Illyrians, through the incompetence of their negotiators, entered the European Economic Community under extremely unfair conditions. Later on, a EEC Council member argues,

Source: Analysis 33, no. 6 (June 1973). Reprinted by permission of the publisher.

Figure 5-3: The first few paragraphs of the Shiner article.

the upper part of the screen, she can try to apply them by looking for the suggested linguistic signs: "many people," "most people." If she were able to see that Shiner's opening sentence ("Many morally sensitive people find themselves faced with the following dilemma.") pointed to a generally-held position which Shiner elaborates in the second sentence, she might make a note of it (See Figure 8), a note to which she could return at any time.

But Janice is confused by the second strategy of "placing the position in the past or recent past." She lacks the discrimination necessary to see that this reading strategy does not apply to these paragraphs. To help her understand how readers can use these strategies more concretely, she can call up models of readers applying these strategies to this paragraph and she can read through transcripts of their protocols.

These protocols are taken both from expert and novice readers and so contain both
HELP MENU

You may:

- look up a word in WARRANT's dictionary
- look at the Nickel article mentioned by Shiner
- look at the Cowan article mentioned by Shiner
- examine how another person read through this passage
- see a list of goals other readers have pursued in this passage
- choose another text to read
- change to another task

Select the option you want.

Figure 5-4: Help menu for the first few paragraphs of the Shiner article.

POSSIBLE GOALS

For the above paragraph, many readers have chosen from among the following goals:

- identify the problem
- identify the generally-held position
- identify the credible opposition
- identify the author's position
- identify the positions the author conceives
- EXPLANATION of goals

Select you choice:

Figure 5-5: A Menu of relevant goals for the first paragraph of the Shiner article.

perceptive distinctions and mistaken decisions. As Janice reads through several of the protocols (protocols which we will have edited to maximize their educational benefit), she decides that one sort of interpretation makes more sense than another.
TASK: An author’s opening moves

EXPLANATION

In the opening few paragraphs, most authors of an argumentative essay (an essay intended to convince the reader of something) will tell you

• what the problem is that they address. Writers seldom write just to see themselves in print. They take a public stand because they feel they have something to say about a problem that concerns or should concern their readers.

• what the generally-held position is. Often—although not always—writers will open their essays by telling their readers what most people think. They intend to get readers interested in a problem they’ve heard about before.

• what the credible opposition is. While not all writers will refer to a generally-held position, they all will refer to a credible opposing position. This is the position they will seriously try to answer. It is usually the take-off point of the rest of the article, the position that contrasts with the writers’ own position. Writers almost always need a credible opposition to justify their writing. If no one disagreed with them, they would not need to write.

• what the author’s position is. By then end of the first few paragraphs, a reader should be able to tell what position the writer is taking on the problem. This position may not be completely explained or fully elaborated until later in the article, but it will probably be referred to and categorized fairly early. This way, writers make an early claim to have something to say about a problem that is new and relevant, and is worth their readers’ time.

• what position the author concedes. Very often there are many positions in a problem that writers do not want to argue about. So they concede them—that is, they allow that they may be true, or accept them as true so that they will not have to discuss them in the rest of their article. This strategy helps a reader to get a clearer picture from the start of the approach the writers are and are not taking.

Figure 5-6: An explanation of an author’s opening moves.

If she wishes to test her judgments against those of expert judges, she can rank order the protocols and then compare her order against those of a panel of judges. Further, she can examine edited snippets of a discussion this panel had concerning the relative strengths and weaknesses of each of the protocols.

Through these various activities with the protocols—reading, ranking, comparing
TASK: Identify an author's opening moves

GOAL: identify the generally-held position

STRATEGIES

Attribute the position to most people:

"commonly," "generally," "traditionally"

"most people," "many people"

Place the position in the past or recent past:

"the belief was . . .," "there has been . . ."

Make the position sound conditional:

"might seem," "it might be," "One could"

Figure 5-7: Strategies for identifying the generally-held position.

NOTE

Shiner, generally-held position

people are faced with a false dilemma

(1) they think that because being black is irrelevant, it does not justify favorable discrimination

and

(2) they think that without favorable discrimination social injustices may continue.

Figure 5-8: Janice makes a note on the generally-held position.

ranks, and examining edited discussions—we expect Janice to begin to develop a finer set of distinctions and a greater sensitivity to the considerations that a reader must notice to understand a text. We believe that she may be better equipped to deal with the complexity of texts she meets in the future, and less likely to ask for the "true" interpretation or the "right" rule for interpretation as so many of our students do now.
Janice returns to the Shiner article and finishes reading through it—calling up relevant goals, looking up a few words in a dictionary, taking a few notes—and then calls it a day.

The next time Janice returns to her work, her needs and history will have changed. She probably will not require the lengthy explanation of the assignment she received on her first encounter; the information shown in Figure 1 will not appear unless she asks for it. She may want to return to the Shiner article; WARRANT will return her to her last reading place if she so requests. Furthermore, her options may have increased; once she makes some notes or starts a draft in one working session, WARRANT will allow her to start her next session with those notes or that draft (See Figure 9).

RETURN TO WARRANT

You may return to the assignment in a number of two ways:
• through to the Shiner article you have been reading
• through the last note you made
• through one of your previous notes
• through a new article to read
• through a task you want to accomplish

Select the type of entry point you want:

The second time into the program, Janice becomes curious about where all this reading and note-taking may be leading her, so rather than entering the assignment through an article, she enters through a set of tasks to be accomplished (See Figure 10).

By looking over the list, Janice sees that she is eventually going to have to read several articles and compare their arguments, but she realizes that she needs to read at least one more article before she can worry about those future tasks. So she selects "read an article carefully" and gets the menu of readings she had received the session before (Figure 2).
TASKS

You may choose to work on any of the following tasks to complete your assignment:

- Read an article carefully
- Reread an article to summarize the argument
- Reason about the argument made in an article
- Compare the arguments in two or more articles
- Write your own critical evaluation of the articles

Select the task you want to work on:

Figure 5-10: The set of tasks for completing a critical reading, reasoning and writing assignment.

With Warrant's flexible representation, Janice's path to a particular frame such as the menu of readings does not restrict the number of options she has open to her. Now that she's in the menu of readings, she has the same options open to her as she did the day before when she entered directly through the texts. At any point, she has five types of information available—readings, goals, strategies, models, and assessments of those models—as well as any notes or drafts she might have made herself. She can enter into all and any of this information through a text, a task, her own note, or her own draft.

So conceived, Warrant will give Janice much of what she needs in order to develop her abilities in critical reading, reasoning and writing. By looking at goal menus, she gets a sense of what options are available. By reading explanations, she gets the information she needs to make a choice among options. By trying out strategies, she acquires procedures to help her accomplish her goals. By examining models, she may develop the discrimination necessary to choose between alternative goals and strategies. And by comparing her assessments of those models against those of expert teachers, she will be able to test the progress of her discriminations.
6. APPENDIX II.

WARRANT: System Design and Implementation

In this appendix, we offer a detailed design for an interactive computer environment, called WARRANT (Writing And Reasoned Reading About Normative Texts), for integrating critical reading, reasoning, and writing.

6.1. Our Design Goals

6.1.1. Flexible Structure

An interactive computer environment to facilitate the process of reading, reasoning, and writing can be no better than the model of the process implied by a particular technology. As case studies of writers using word-processors have suggested, technologies sometimes impose unduly rigid and confining models of writing that can strait-jacket writers. Most existing writing programs, built on a limited technology, impose a rigid structure on the writing process. For example, in the recently marketed WRITING IS THINKING program a student must always create a sentence outline before beginning to compose. Many composition teachers and researchers would question the program documentation's claim that WRITING IS THINKING sets the writer "within firm limits that free him to be creative...", and would object to the rigid model of the composing process that this software imposes. With recent advances in microcomputer technology that have yet to penetrate the classroom, however, the limiting factor in developing computer tools for readers and writers is now less the technology than our lack of knowledge about the reading, reasoning and writing process, and how that process develops as students mature.

Research in the composing process suggests that experienced writers vary widely in how they write. Some writers jot down an idea, explore organizing frameworks, and then flesh out the idea with detail and polish. Other writers generate many ideas and only then search out organizing frameworks. These observations suggest the importance of developing an environment that allows writers to move freely through the many possible paths of the writing process.

On the other hand, teachers of writing and researchers studying student writers observe that students can easily become overwhelmed by the enormous number of options the writing process presents. Many teachers and researchers agree that
students, as opposed to experienced writers, require a degree of external structure in order to gain better control over their writing processes, and that as they gain control, the externally provided structure should be withdrawn.

Our response to this problem—that students learning to read, write, and reason need adjustable mixtures of freedom and external control—is to set a design goal of flexible structure. The design specification of our WARRANT system includes facilities that will help us explore how much structure to provide. In the default case, students will be able to pursue tasks in any order. However, we will be able to adjust the system so that students' access to tasks is either blocked or conditional on their having completed other tasks. In this way, we can adjust our instructional guidance so that it is as structured and intrusive or as environmental and unobtrusive as we like.

This facility will make WARRANT an environment that can be adapted to both more experienced and less experienced writers. No less important, it will make our project a useful tool for researchers who wish to study the development of reading, writing, and reasoning processes.

6.1.2. Integration of Tools

Early computer programs to facilitate writing provided a single tool, usually concentrating on a single subprocess of the process of composing. One of the earliest and best known examples of a software tool to facilitate the process of composing is Hugh Burns' set of programs for invention brainstorming.26 As useful as Burns' program might be, Burns did not integrate his brainstorming tool with any other aids for writing, reading, or reasoning.

Similarly, early tools that were developed for revising text rather than for producing it were not integrated with other writing aids, not even text-editing tools. For this reason, users of WRITER'S WORKBENCH,27 the earliest and still the most advanced tool for checking texts for stylistic weaknesses, cannot edit a text interactively with WORKBENCH's recommendations. Instead they must print a complete list of these recommendations and then search through their original manuscript to see whether the recommendations apply.

Early programs to facilitate composing did not integrate tools for two related reasons. The first is money. With earlier technologies, it was prohibitively expensive, both in memory space and response time, to provide integrated
environments. The second reason is lack of experience. As writers gained experience using computers interactively, they began to see the advantages in having an integrated set of tools available, insights that were not obvious earlier.28

More recent efforts have partially integrated multiple writing tools within a single environment. WANDAH, recently developed by Von Blum at UCLA for the IBM PC, provides pre-writing, text-editing, and revising aids within a single environment. However, limitations in the IBM PC's memory and screen-size prevented Von Blum from a fully integrated design. For example, when students enter WANDAH's revising module, WANDAH can highlight problematic constructions, but due to the computer's memory limitations, students cannot use WANDAH's text-editor to change the highlighted constructions from within the module. Like users of WRITERS WORKBENCH, students must get a hard-copy print-out of the problems, then go to WANDAH's text-editing module to search for the problems and revise them.

A major premise of this proposal is that writing requires a highly integrated environment of tools that include tools for reading and reasoning as well. Therefore, a major design goal for WARRANT is that the tools we build to facilitate these skills be highly integrated. In our design specification, the text-editor is available at all times. While reading, students will be prompted to reason and write in response to what they read. While writing, students will be prompted to reason and use the writing that they generated during reading. And they will be able to reread an assigned text without having to suspend their writing activities.

There is, of course, a danger that an environment that is flexibly structured and highly integrated in the way we have suggested will also be more complex, more difficult to use, more expensive, and thus less accessible than tools that are rigid and segmented. Given this danger, we need to consider two other design goals: user-friendliness and a high potential for dissemination.

6.1.3. Easy to learn and to use

Inexperienced writers and readers are burdened with heavy cognitive demands, and certainly don't need the added burden of working with a system that is not easy to learn and to use.29 Opinions about the user-friendliness of different computer systems abound, but there has been little scientific study of the system properties and user abilities that make a system easy to learn and to use.30 Despite the lack of detailed research into "user friendliness," some principles have emerged which have guided recent improvements in interface design.31 Perhaps the most important
principle to emerge is the maxim, "get to know the user," that is, test critical parts of a system on representative users.

For WARRANT, our initial group of users will include both students and their teachers. The system software on which we will be building WARRANT both adheres to and improves upon successful designs. The system software is also undergoing testing in a joint project involving Carnegie-Mellon's Information Technology Center and Communication Design Center staffs. We are committed to testing WARRANT as we develop it.

6.2. System Architecture

Before describing the dissemination potential of WARRANT, we will describe several important features of the system hardware and software on which we will be developing WARRANT. Without these resources as a foundation, it would be impossible to build a program like the one we are proposing.

6.2.1. System Hardware

We propose to develop our program on IBM Peaches. The Peach is a completely self-contained single-user computer which can communicate with other computers over a local area network. It is built around a 32-bit microprocessor, with 1 megabyte of memory, two 10 Mbyte disks for local storage, a keyboard, a mouse pointing device, and a high-resolution bit-mapped display.

These hardware components represent a set of planned capabilities based on the projected use of personal computer technology for educational computing. For WARRANT, the 32-bit architecture allows compact, efficient coding of many basic functions and is essential to good performance on such tasks as labelling and sorting notecards. The megabyte of memory, roughly twice the maximum memory available in a current IBM PC, will allow us to integrate text-editing and revision guides. The two 10 Mbyte disks for local storage coupled with network communication for central storage will allow on-line storage of readings, expert protocols, student performance data, etc. In addition, local area network communication will provide student writers with a context and audience for their writing, since they will be able to share their work with teachers and peers.

A mouse pointing device allows users to move the cursor rapidly over a large, free-form display screen with enough precision to allow easy selection of even
From a writer's point of view, the most important hardware element for our design is the high-resolution bit-mapped display screen, with a 10 by 17-inch display. Until recently, memory costs were too high for bit-mapped displays to be implemented in small, inexpensive computers. However, memory cost is now low enough to make such displays feasible even for home computers—the recently introduced, low-cost APPLE Macintosh has one.

Indeed, APPLE's marketing campaign for the Macintosh has helped to educate the public on the advantages of high-resolution bit-map display devices. These devices (1) make the screen easier to read by allowing graphic design principles requiring boldface, italics, and different font families and sizes to be readily implemented; (2) increase application program's capabilities by allowing many "windows" of information per screen; and (3) enhance the writers' ability to edit by displaying the text on the screen in the way it will appear on a hard copy. (This capability is standardly called "what you see is what you get" editing.)

Each of these three advantages of bit-mapped displays is important to the design of our system. Let us consider each in turn.

6.2.1.1. Improved Screen Readability

Integrating reading, writing, and reasoning will require students to read articles, to receive instruction and prompts, to study protocol transcripts of other writers, and to study assessments of these protocols all on-line. For this reason, we consider first-rate screen readability essential to our design efforts.

Under the sponsorship of Carnegie-Mellon's Information Technology Center, our Communications Design Center (CDC) has recently started to investigate the relative readability of printed text, conventional CRT (Cathode Ray Tube) displays, and bit-mapped displays. Preliminary results of research currently underway in the CDC suggest that students find both bit-mapped displays and printed text significantly easier to read than text on CRT displays. Based on previous research in graphic design, we expect to find that improved readability for bit-mapped displays stems from the improved graphics techniques. Unfortunately, with CRT displays, most designers of educational software have had access only to the most elementary graphics techniques such as reverse video and blinking.
6.2.1.2. Multiple Windowing

Bit-mapped displays permit a screen to be divided into any number of display windows, of any size and shape the student chooses. Conventional CRT displays, by contrast, allow students to split the screen only into two horizontal windows of twelve lines each. Why do we regard this more flexible "windowing" capability essential to our project? We will want students to be able to view different pages from the readings at the same time, view different notes in several windows, view some of their notes while composing, and view teachers' and peers' comments while they are revising their texts. Thus we will need a large display, with flexible and easy-to-use window management facilities.

6.2.1.3. What You See Is What You Get Editing

Students, as well as professional writers, typically find it frustrating to print a hard copy of a document from a word processor only to find it filled with many formatting mistakes that did not show up on the screen. "What you see is what you get" editing eliminates this frustration by allowing students to see their text on the screen as it will look when it is printed. Of the three features we have discussed relative to bit-mapped displays, this feature is the least essential for our project. Nonetheless, we expect our system to receive wide use in freshman and sophomore courses among students who have not had a great deal, if any, experience with text-editors. We think this feature will eliminate much student frustration. In addition, we hope that our proposed program will eventually be used by more advanced students such as the technical and professional writing majors. Part of their coursework involves studying techniques of graphic design which improve written communication.

6.2.2. System Software

The underlying operating system is Unix40 4.2 Berkeley Standard Distribution (BSD). All systems software is written in the C programming language.41

Since Unix is notoriously difficult for novice users to learn and for infrequent users to remember, it is important to distinguish the underlying operating system from the user interface. The user interface, will be modelled after APPLE's LISA "desktop metaphor," a design that makes excellent use of recent advances in human-machine communication.

The implementation of the program we are proposing will be divided into two parts: an integrated text-editor/window-management program and a networked menu-
selection system. We propose to use the text-editor/window-management program built by James Gosling of Carnegie-Mellon's Information Technology Center. Since it is key part of our design, we will describe it in some detail.

6.2.2.1. The text-editor/window-management program.

All applications programs which interact with the user will use Gosling's text-editor/window-management program to mediate interactions, so the entire system will have a uniform interface.

The window manager. The window management program allows windows to be displayed as "tiles" on the screen. There can be an arbitrary number of tiles, of any shape and size. Gosling's design departs from the Xerox's Star or Apple's LISA or Macintosh window management in one crucial respect. In the Xerox and Apple systems, windows overlap, so that one window can hide other windows. Overlapping windows, however, can create a problem. In part, windows were developed so that everything that is relevant to a task would be visible on the screen. The screen can then relieve the load on a user's short-term memory by acting as a memory aid. When windows hide other windows, this purpose is defeated. As on a real desk-top, documents can become temporarily lost under the pile, leading to time-consuming searches. By creating a window-management system based on tiles rather than overlaps, Gosling avoided this problem.

Pop-up menu system. Gosling's program does use overlapping windows, however, for pop-up menus which hide a column of text briefly. When the user presses the button on the mouse, a menu of available commands "pops up" on the screen. By continuing to hold the button down, the user can move the cursor to the desired menu item. These menu items themselves can have sub-menues which pop up as the cursor moves over them. When the user releases the mouse button, the item on the menu is selected, the corresponding command is executed, and the menu disappears. Pop-up menus are used for the small set of commands that can be used throughout the system, e.g., help, "cut and paste," copy, etc.

On the Xerox Star, such commands were implemented with special function keys on the terminal keyboard, a design solution that has two limitations. First, it necessitates a special keyboard with appropriately marked function keys which limits the transportability of a program which uses them. Second, unlike the QWERTY keyboard, there is no standardization of function key placement or assignment, and with anarchy reigning, users must learn which keys control which functions for each
separate application. Gosling's menus also differ from those used on the APPLE LISA and Macintosh, where such commands are implemented with pull-down menus, displayed horizontally across the top of the screen. However, horizontal layouts only work well with a small number of menu options. Typically, each menu option must be limited to one or two words, or there won't be enough room for all the choices. A vertical layout is a better choice when more room is needed.

Scrolling and Sharing Information. Like the Xerox Star and Apple LISA, information or text can be scrolled horizontally and vertically within any window, and the user can select any amount of text from a document displayed in one window and copy it into a document displayed in another. For our application, these are important features, since we want students to have easy access to the full context of texts, and to be able to move information from previously generated notecards into texts they are drafting without having to retype it.

The Base Editor. All programs which interact with the student use the base editor to mediate interactions, so the entire system has a uniform editing interface. For example, the student will interact with the mail program, bulletin boards, and other system programs, including WARRANT, through the editor. Thus students will only have to learn one editor for the entire system.

The editor is similar to Xerox's Bravo or Apple's LisaWrite. It is an interactive editor and formatter, which means that the user can edit and format a document at a terminal and see the results immediately displayed on a full page, high resolution screen. Although it does not maintain an exact replication of what will appear on the printed page, it maintains a close approximation. Bold and underlined text are displayed appropriately, indented copy is indented the proper number of spaces, centered copy is centered on the line, etc. The conceptual model for the formatting is based on SCRIBE, a "batch" or post-editing text-formatter that was designed to make document production easy for the non-expert and now has widespread use.

Command Structure. Commands in Gosling's text-editor are structured like English cleft-imperatives, consisting of a noun followed by a verb (e.g., the rope—hold it). The nouns of interest are typically regions of text, e.g., characters, words, sentences, lines, paragraphs, etc. These can be specified by pointing to the text using the mouse or by typing commands at the keyboard. Functionally, the verbs sort into three categories: user aids (help, undo), editing (erase, cut and paste, previous), and formatting (make title, boldface). The editing and formatting commands apply to the
Ease of Learning and Use. The design of many computer programs assumes that there is a single ideal system which can satisfy all users. However, research suggests that there are two very distinct groups: (1) the sophisticated, frequent user, or expert, and (2) the unsophisticated, infrequent user, or novice. These groups differ in both their competence and their demands for how the system should operate. Often a system that is easy for novices to learn is unsatisfactory for experts to use. For example, novices find a menu easy to use because it allows them to see commands rather than having to remember them. Many experts, however, find menus inefficient for accomplishing their goals. User interaction with Gosling's editor was designed for both groups of users. The user can use the mouse at any time to pop-up a menu to the screen and select a command. Alternatively, the user who knows the command can type it at the keyboard.

Modeless Interaction. A mode is simply a way of behaving, so "modeless interaction" is a bit of a misnomer—for no system can be truly without modes. However, many computer systems use more than one mode because there are too few keys on the keyboard to represent all the available commands. In response, the designers add more than one mode, and the interpretation of the keys then depends on the mode that the program is in. Thus, many text-editing programs have two modes, "edit" and "insert." When in edit mode, pressing a "d" on the keyboard means "delete;" when in insert mode, pressing a "d" means "insert a 'd' into the text. Modes often cause trouble when a familiar command from one mode has an unexpected result in another. Therefore, there has been much recent interest in "modeless interaction," a prescription to try to minimize, if not eliminate, modes. Gosling's design follows the prescription for modeless interaction, and does not have separate modes for inserting and editing.

6.2.3. Why We Chose this System Hardware and Software

The IBM Peaches, at least initially, will not be inexpensive. They are state-of-the-art personal workstations with more memory, better quality display monitors, and more central-processing power than are presently available in lower priced, more widespread personal computers (e.g., IBM PC, APPLE MACINTOSH).

Although we have discussed the system hardware and software before detailing our design for WARRANT, we chose the hardware and software with our design goals and
design in mind. Four considerations determined our choice: (1) WARRANT requires the hardware and system software capabilities that this system provides; (2) by choosing this system, we can build WARRANT on existing tools rather than starting completely from scratch, saving at least a year in the timetable for our project; (3) the system will have widespread use at Carnegie-Mellon, so WARRANT will be more likely to be widely used on campus; and (4) WARRANT’s dissemination potential is high if we develop it on this system.

6.2.3.1. WARRANT’s requirements

Many software design efforts begin with hardware specifications, then pare down the ideal software system to something that will run on the hardware. We began with our software design, and then consulted with several computer scientists\textsuperscript{48} for hardware, feasibility and cost estimates.

As we indicated in previous sections, we determined that developing on more widespread, less expensive machines would incur significant sacrifices of our design goals. Moreover, according to current projections, machines with the capabilities of Peaches will be available to educational institutions inexpensively by the time our proposed project has reached the stage when it is appropriate to disseminate WARRANT widely. A few years ago, the combination of marvelous capabilities and low cost seen today in an APPLE Macintosh was undreamt of by many educational institutions.

Carnegie-Mellon is committed to helping its faculty develop educational applications on advanced technology so that educational goals do not need to be sacrificed to limited technologies and so that development efforts need not be outdated before they are completed. We are fortunate. But, as prices for small machines continue to drop at a remarkable rate, we believe that many other schools across the country will soon be able to share in the benefits of more powerful technology. In a presentation to the U.S. Department of Education and Office of Educational Research and Improvement’s Research Conference on Computers in Education, Raj Reddy observed that the price of computing power will continue to drop, bringing phenomenal computing power, far above what we will be using, within reach of educational institutions:

...what computer technologies will permit us to achieve our educational goals, in terms of computational power and economic feasibility? If we consider the power of presently available personal computers that cost approximately 100 dollars, we can estimate the computational power we might be able to get for under 100 dollars in the next decade. If I were to
say we could have a super computer somewhat like the Cray-1 for under 100 dollars, you might think that it is too farfetched; I thought so too when I was trying to make such an estimate. But I have extrapolated the technologies and it indeed appears possible to have that kind of affordable computation power in the very near future. ... Given the current level of computer sophistication, I anticipate that we will have an affordable Cray within the next decade: a 100 MIPS (million instructions per second) processor; a million characters of random access memory; and four million characters of program (read only) memory. ... Extrapolating by the same trend, we could have a 100 MIPS computer by 1990.49

6.2.3.2. Building on Existing Tools

Developing, implementing, and testing a reliable, well-designed text-editor/window management program represents a major undertaking, and even a scaled down version would add at least a development year to our proposal. Gosling's editor was specifically designed for use by application programs such as WARRANT and, by using it, we will be able to focus on developing more specialized tools for reading, writing, and reasoning.

6.2.3.3. Increased Likelihood for Use on Campus

The Peaches are prototype machines and may never have widespread use on campus, or for that matter, anywhere else. However, system software and applications program developed on the Peaches will be compatible with whatever personal workstation that is eventually deployed widely on campus. Developing WARRANT on the Peach will thus increase the likelihood that WARRANT will receive widespread use on campus.

6.2.3.4. Good Potential for Dissemination

The Peaches may be mass marketed and become widespread, but even if they do not, WARRANT will still have good potential for wide dissemination. As we argued above, machines with the Peach's capabilities will be available by the time WARRANT is ready to be disseminated widely. Furthermore, the operating system for WARRANT, Unix 4.2 BSD (Berkeley Standard Distribution), has been adopted as the standard for the Consortium of Universities of which Carnegie-Mellon is the head. As a result, WARRANT will have high potential to be shared rapidly with members of the consortium. In addition, Unix itself is a highly transportable operating system that runs on machines of many different sizes, so WARRANT will have the potential to be shared widely.

Although hardware and software concerns are often raised in discussions of a program's potential for dissemination, we believe that an equally important concern
is the instructional quality of the program and how easy it is for teachers to adapt the program to the needs of their students. Post-secondary education covers a wide territory. Students vary tremendously in their educational backgrounds and goals. It is impossible for a single project to address the needs of all these students. If the design of the system allows teachers to easily adapt the program for their students, however, the potential for dissemination of the program is greatly enhanced.

Because our design goals call for flexible structure and integration of tools, WARRANT will have good potential for adaptation. Although it is beyond the scope of our three year project to provide an authoring system for teachers to make it easy for them to modify WARRANT any way they like, To build such an authoring system would require a 1-3 year effort beyond the development of WARRANT—an effort we firmly intend to take up. The same tools that will aid us in formatively evaluating WARRANT and exploring the best ways to structure curriculum guides will eventually enable teachers to modify the instructional guides or create new ones. Since a text-editor is integrated with the entire system, teachers will eventually be able to use the same text-editor to create and modify instructional guides that they and their students use to interact with the WARRANT for reasoning, writing and reading tasks. Thus, teachers will be able to control significant portions of the instruction without having to know details of computer implementation or programming. Since we will be keeping records of students' use for formative evaluations, teachers will eventually be able to use similar records to guide their decisions to modify WARRANT.

6.3. Our Proposed Program

In this section, we will describe the design of WARRANT. Before we discuss our design, however, we would emphasize that modifications may result based on our task analysis and on formative evaluations.

6.3.1. Design Methodology

6.3.1.1. Task Analysis

One of the crucial elements of this proposal is the task analysis that we propose to do. The task analysis involves establishing, on a firm empirical basis, the goals, strategies, tests, and schedules of experts and novices as they work through a reading, reasoning, and writing task. This analysis will help us refine what we already know about pedagogy from our own teaching experience and research in
order to build a rich learning environment for students.

6.3.1.2. Formative Evaluation

Part of our design methodology is to build on existing tools, so that we can more easily implement a pilot, prototype system, and then test our ideas. We know from hard experience in designing instructional programs that our initial system will not be our final one. Systems evolve. We have implemented systems that share features with the program we are proposing. Neuwirth and Kaufer implemented a program that shares many features with the WARRANT. Based on feedback from users and expert collaborators, we built improvements that were incorporated into a later version, and into the present proposal.52 We believe in prototyping and formative evaluation as an integral part of design methodology.

With those caveats aside, we discuss our design for WARRANT with the understanding that modifications may result based on our task analysis and on formative evaluations. The following discussion is organized around the set of tools that WARRANT offers to students during their process of reading, writing, and reasoning.

6.3.2. Tools for Reading and Reasoning

Research suggests that two important differences between expert and novice readers are how actively they read and how well they are able to draw on discourse schemas to integrate what they read. To read with deep understanding, readers must do more than scan their eyes across a page, memorizing the contents. Active reading, reading with understanding, involves responding to the text, participating in a mental dialogue with the author—if you will, “writing” the text that is being read.53

We propose to develop a tool which will prompt novices to give the types of responses to texts that we observe in experts, to provide novices with feedback on their use of these prompts, and to use their responses to the readings during their writing.

6.3.2.1. On-line Reading: Interactive Texts

We propose to store assigned readings on a value issue on-line and to implement three ways to interact with the readings: (1) reading an article actively, (2) reading an article to summarize its contents, and (3) relating the argument in one article to those in others. For the purposes of exposition, we will describe the system design features for the first option, reading an article actively. The other reading tools will
have similar features.

When a student chooses to read an article actively, he or she will see the article displayed by Gosling's text-editor in a window on the screen. The student will have complete access to the entire article, but one paragraph, called the "current" paragraph, will have a special status and will be surrounded by the outline of a box. This highlighted paragraph is the paragraph the program assumes the student is focusing his or her attention on. When the student first enters the program, the current paragraph will be the first paragraph. By selecting from a menu, the student will signal the program that the paragraph pointer should move ahead to the next paragraph, or back to the previous paragraph. Again, the student can always read other parts of the article without moving the current paragraph pointer.

6.3.2.2. Prompts: Modelling Goals and Strategies for Reasoning

In addition to highlighting the current paragraph, a menu will display a set of goals that other writers have pursued while reading the paragraph. The menu could include such prompts as "clarify the author's position," "relate a statement to the argument structure," "give a counterexample," etc. By providing students with menus of goals, we expect that students will be prompted to apply more sophisticated responses to the text. But they might not have strategies for pursuing those goals. By selecting a particular goal from the menu, however, they will be able to access general strategies with accompanying tutorials, concrete models of writers pursuing those strategies, and assessments of the success or failure of those strategies for the selected goal.  ^54

6.3.2.3. Note Cards: Integrating Reading and Writing

When the student selects an item from the menu, a window with a "note card" will appear on which the students can record a response. Unless the student requests otherwise, the card will automatically be filed by the system under the indices for the relevant article, paragraph, and heading from the menu. Students can also provide their own label. These indices will provide a way for students to retrieve the notecards. For example, they can ask to see all their notes from James Nickels' "Discrimination and Morally Relevant Characteristics" or all their notes for the issue of "minimally skilled vs. skilled labor," an issue that they might have taken notes on in response to several articles. Notecards will have a default organization associated with them, by author and order in the relevant reading passage.

However, students will be able to impose their own permanent ordering of
notecards for ease of retrieval. In addition, students will be able to form an order that is not based on automatic retrieval, but is based on their browsing through the cards and putting them into an order. Notecards will have links back to the articles, so that from any note, they can access the article. We expect to achieve significant integration of the processes of reading and writing from this mechanism.

6.3.2.4. Monitoring: Modelling When to Pursue a Goal

We expect students to learn a great deal about when to pursue goals by observing models of experts. We will also explore the possibility that monitoring students' pursuit of goals and providing them feedback about it will facilitate their learning.

As the student is selecting from the menu options for responding critically to an article, the system will be recording their selections and comparing them to a record of what our experts and novices did when reading the current paragraph. When the student chooses to go on to the next paragraph, the student will be given feedback, when appropriate. For example, if the majority of the experts found a statement in the paragraph problematic in some way, and the student has not responded to it, the system can notify the student that he or she has overlooked a point that experts considered important, and ask the student to respond to it.

Similarly, if the majority of novices misinterpreted a statement, the system can prompt the student to give an interpretation, then compare that interpretation to the novices, together with a discussion of why it is a problem.55

6.3.2.5. Do Goals and Strategies Generalize to Other Texts?

To test the generalizability of the goals and strategies we use for our menu system, we shall test them as prompts in other writing environments. More specifically, we shall give them to students reading other articles of similar and different genres.

6.3.2.6. Limitations and Long-range Goals

In a more advanced system, prompting would be more tutorial, based on techniques of intelligent tutoring systems, such as GUIDON, an intelligent tutoring system that tutors medical students in the diagnosis of infectious diseases.56 For example, from the protocol analysis of experts, we would derive an AND/OR tree of goals and rules that the experts used. This would provide the basis for a goal-directed discussion that the program could carry on with the student. Like a human tutor, the program could actually analyze students' responses. Constructing an artifically intelligent
environment capable of more human-like tutoring is a long-term goal, but in a task as complex as ours, it is too ambitious. Our proposed system is tutorial in nature, but more limited in scope, and although it will not display the sensitivity we would like, it will help us explore and lay the groundwork for a more ambitious system. Because the system saves student responses, we can use it as a tool for exploring the feasibility of providing more sensitive feedback.

6.3.3. Tools for Writing & Reasoning

Computer text-editing programs provide a set of very general, but low-level tools to anyone who works with text—copy editors, typists, or writers. A typical text-editing program allows a person to insert, erase, and “cut and paste” words, sentences, paragraphs, even entire sections of a manuscript without the tedious copying and recopying required by more traditional technologies. By using a well-designed text-editor, a person can carry out plans for writing and revising easily.\(^57\)

Writers, however, do much more than carry out plans. They must also produce those plans, a task that often involves all the components of the reading, reasoning, and writing processes.\(^58\) In our proposed system, students will compose their papers using the text-editor, but they will also have access to goals, strategies, models, and assessment of these models for planning and organizing their paper which we will build, as well as an environment to aid them in the process of revision.

6.3.3.1. Planning

Like the reading environment, students will also have access to goals, strategies, models and assessment of models. In the writing task, these will include problem structure and purpose, audience knowledge and characteristics, and the writer’s representation of a self to be conveyed. In responding to prompts, students will have their notecards from their reading available and will be able to create new notecards.

6.3.3.2. Organizing

We expect that students, using the reading and planning prompts, will be able to generate a wealth of ideas, but that these ideas may be in the form of key-words and brief notes, and that the order will be the order in which they were generated. We will develop a set of prompts to aid students in organizing their notes, using both our observations of experienced writers and the technique of issue analysis.\(^59\) The facilities for searching, browsing, and reordering notecards will be integrated into our prompting system.
We will also explore the pedagogical effectiveness of using the critical reading prompts as means for prompting students to elaborate their texts for a reader.

6.3.3.3. Revising

Unlike experienced writers, student writers tend to revise mostly at the word and sentence level. But student drafts are not greatly benefited by attention to low-level editing details when major problems remain in their text. In an insightful article, Linda Flower identifies a cognitive basis for many student writing problems—students tend to write egocentrically. Prose that communicates needs to be sensitive to the needs of the reader. We propose to develop this sensitivity by providing a context and audience for student writers, a context in which students can comment on each others' papers.

In WARRANT, a teacher or peer will be able to tag parts of a text with functional/structural labels (e.g., function = comment, structure = paragraph). The comments will be stored as annotations to a text, but will not be visible to the author unless the author asks to see them, or part of the instructional guide displays them. In the former case, the author can specify the level he or she wishes to see. For example, find all the comments that have to do with paragraph development, and display each one. The relevant part of the text, together with the comment in another window, will appear. The text will be displayed using the display management facilities of a text-editor, so the whole text will be available while reviewing comments.

Similarly, the person making a comment (either an instructor or peer), makes the comments from within the text-editor, and the prompts for critical reading will be available while creating comments.

6.3.4. Tools for Managing the Environment

6.3.4.1. Curriculum Guides

WARRANT will use menus to display options to users, and to allow them to choose an option quickly and easily. To choose an option, a user selects the option with the mouse pointing device, or types its associated letter, number or symbol. When a user chooses an option, a new display appears, perhaps itself displaying further sub-options relating to the option just chosen. In addition to moving to the new display, choosing an option may cause the system to perform some action. For example, if a student would select the option to "Read an article actively," a new display
showing the articles that are available would appear. In addition, the system would perform an action, recording that the student is "reading actively" in a data structure used for communication to the display manager for the text-editor/window-management module.

6.3.4.2. Curricular Prerequisites

The display of options will be conditioned on a data structure which we will refer to as "curricular prerequisites." The display processor and selection processor will check curricular prerequisites. Options which are currently available to the student will be highlighted; those options not currently available will be visible, but a graphical cue will indicate that they are not available (e.g., displaying gray text on a gray background, instead of black on white, which is used in APPLE's LISA and MACINTOSH systems). There can be multiple curricular prerequisites. The one in force, if any, will be found in a student's record, or inherited from the student's class record. The prerequisites will be maintained in a form that can be text-edited.

A student record will be maintained, which will record a student's progress through the prerequisites. The display and selection processing managers will also check this guide. For example, if the student has completed the reading and summarization guide, then the selection processor will allow access to the invention guide; if the student has completed the invention and organizational guide, then the processor will allow access to revision guides.

For writers who are not given curricular prerequisites, multiple windows will allow them to keep temporarily suspended goals active while they pursue another goal. For example, if a writer moves directly to invention without doing a summarization, but an invention prompt suggests that a summary of the reading is necessary, the writer can keep the invention goal active in one window while going through the summarization activities in another.

The results of doing any task, for example, a summary produced from following the instructional guide for summaries, will always be accessible for use in other tasks.

6.3.4.3. Statistics Gathering

To aid us in our formative evaluation, WARRANT itself will help us gather data about how students use it. We are planning two types of data-gathering facilities called dynamic and log statistics. Dynamic statistics will be gathered whenever a student is interacting with WARRANT. The mechanism will record a summary of all
student actions. These summaries will be stored in a central file when the student exits the system. In addition to providing us with valuable information about how students are using WARRANT, they will form the basis for allowing students to exit WARRANT and reenter without losing their places in the tasks they are pursuing. Whereas dynamic statistics are always kept, log statistics will be gathered only upon request. They will be much more detailed, consisting of a complete history of keystrokes, system response, and a time stamp for each action. We will use external statistical packages to analyze both types of records.

6.4. Plans for Formal Evaluations
We are not proposing to do a "summative" evaluation in the three years of the project. We will seek additional funding to carry out such a study. Our central question is "Will WARRANT be able to accommodate and even enhance students' composing processes?" Our evaluation will address this issue by studies designed to answer the following questions:

- Do students produce essays judged to be of better quality using the system than they produce using more traditional technologies such as paper and pencil or typewriters? In what ways is the environment effective and why? In what ways is it ineffective and why?

- Do students learn from the environment? If they produce better essays when using the environment, do they continue to do so if they return to another method?

- Do writers consider the curricular guidelines adequate, or are there important goals and strategies which cannot be expressed in this framework? Do writers like the environment?

- Do more experienced writers use the system differently than less experienced writers? If so, are there better ways to structure the environment for different groups?
7. APPENDIX III.

Project Commitment and Support Documentation

7.1. Institutional Support Units

Inter-University Consortium for Educational Computing (ICEC)

In 1983, Carnegie-Mellon founded a consortium of universities and colleges committed to the integration of information technology into education at the university level. The Inter-University Consortium for Educational Computing (ICEC) has been funded by a three-year, one million dollar grant from the Carnegie Corporation of New York with Douglas van Houwelling and Jill Larkin, both of CMU, as principal investigators. The ICEC staff resides at CMU.

Membership in the Consortium was completed in December of 1983 and includes the following universities and colleges:

- Brown University
- California State University, Northridge
- Carnegie Mellon University
- City University of New York
- Columbia University
- Cornell University
- Dartmouth College
- Howard University
- Iona College
- Mills College
- Rensselaer Polytechnic Institute
- Southwestern College (Chula Vista, California)
- Stanford University
- University of California, Berkeley
- University of Wisconsin, Madison
as well as EDUCOM—a consortium of 400 colleges and universities committed to the use and management of information technology in higher education, and the Research Libraries Group, Inc.—an organization of research and university libraries devoted to the use of computer technology.

The principal goal of the Consortium is to develop high-quality educational uses for the powerful individually-used work stations as they become available over the next few years. Among the tasks involved are:

1. Establishing standards for target hardware configurations and operating systems appropriate for higher education applications.

2. Establishing standard formats, languages and design parameters as well as evaluation procedures for software development.

3. Evaluating educational software developed at member institutions as well as quality software from elsewhere for distribution to members.

ICEC will provide our project with consultative resources for assessment of our system design and implementation, as well as a network for dissemination of our system. The central role of CMU in this consortium will assure dissemination of news of our project and progress to the other major centers of educational software development.

Center for Design of Educational Computing (CDEC)

The goal of the Center for Design of Educational Computing (CDEC) is to provide a productive intellectual working environment for faculty who are developing educational computing at Carnegie-Mellon University. CDEC is currently supported from three sources: internal CMU funding, a grant from the Sloan foundation for the development of educational computing in the liberal arts, and a three-year grant to ICEC by the Carnegie Corporation of New York. In addition, two proposals have been submitted, one to the National Science Foundation for development of educational computing in mathematics, science and engineering, and one to the Annenberg Foundation for data base management.

CDEC’s current activities include

- helping to establish useful collegial relations at CMU,

- providing information and intellectual stimulation through seminars, a newsletter and a library of quality software,
• providing access to a variety of dissemination vehicles for quality software

CDEC will provide us with added internal peer and technical review of our system, a conduit for dissemination, help in securing follow-on funding (for example, for formal evaluation of the system, once it is implemented), support in obtaining the advanced IBM workstations for porting our system from the prototype Peach workstation on which it will initially be developed, and consultation on porting the system to other machines for purposes of dissemination beyond CMU and the ICEC.

Communications Design Center (CDC)

The Communications Design Center at Carnegie Mellon University was established in 1978 by an NIE grant to encourage interdisciplinary approaches to the solution of visual and verbal communication problems. Now administered out of the Dean's office of the College of Humanities and Social Sciences, the CDC is directed by John R. Hayes of the Psychology Department. The Center draws on the skills of a distinguished group of faculty members in the departments of English, Psychology, Design, and Computer Science, and on a highly skilled staff, all of whom have published in the document design area.

For the past few years, the CDC has been actively involved in studying the effects of communication (including instruction) on technology and vice versa. The CDC has an exclusive contract with CDEC to write documentation for all educational software used on campus. We will use the expertise of the CDC to write and assess user documentation for the WARRANT system.

The Writing Center

The Writing Center represents the arm of the CDC concerned with evaluating and developing technological aids for writers. The Writing Center was established in June of 1983 from an internal grant and received, in October of that same year, a grant from the Buhl foundation to evaluate writing software. Chris Neuwirth directs the Writer Center.

The Information Technology Center (ITC)

The Information Technology Center was created in 1983 with funding from IBM to administer the design and development of the IBM workstation that will be distributed throughout the campus by the late 1980's and the communications network for these and the other powerful workstations that will pervade the campus. Other ITC projects are at the forefront of information technology and management. The
ITC is developing the prototype Peach workstation and its advanced resident software that we will employ for developing the WARRANT system. The ITC will install and maintain the workstations dedicated to our project. It was with the cooperation of the ITC that we learned of the capacities of the Peach workstation, reviewed its prototype facilities, and were thereby able to conceive the ambitious WARRANT system, in advance of the rest of the campus community. Consultants at the ITC helped us to assure the feasibility of these ambitions. Systems designers at the ITC will be responsive to design innovations for their developing software that our project generates: our project will be one of a few campus 'labs' for testing and enhancing the resident software and facilities of the ultimate IBM workstation.
7.2. Project Personnel

Preston K. Covey is Head and Associate Professor of Philosophy at Carnegie-Mellon University. He designed and for the last seven years has directed the college core curriculum course in philosophy, Philosophic Methods and Social Values. He developed and for the past five years employed an innovative package of computer-assisted instruction programs in this course, to teach formal logic and its application to argument analysis and basic tasks of philosophic inquiry. This package, initially developed under a previous FIPSE grant, includes the only extensive curriculum and heuristic-guidance programs for analytic reasoning available on microcomputers. He has written a textbook and published articles on heuristics of philosophic analysis, their underlying pedagogy and methodology. He has recently completed a study of the state of computer-assisted instruction in philosophy, commissioned for a forthcoming volume of the Modern Language Association on computer-assisted instruction in the humanities. For a Lilly Post-Doctoral Teaching Award Fellowship he conducted a protocol study of students’ reasoning processes using his heuristic-guidance computer programs for argument analysis, and consequently is familiar with methods of protocol research. For his innovative approach to teaching philosophic methods and analytic methodology, he received the college teaching award. He presently is Project Leader of the Computer-Assisted Reasoning Group of the Center for Applied Logic, a joint venture being organized between Carnegie-Mellon and the University of Pittsburgh. He teaches a graduate seminar on Logic, Epistemology, and Rhetoric for CMU’s Rhetoric Program, in which he’s developed a conceptual and methodological framework that will inform the WARRANT system. He has also been actively involved in importing philosophic methods into interdisciplinary policy studies he’s directed funded by the National Science Foundation and the National Endowment for the Humanities.

David S. Kaufer is an Associate (effective fall, 1984) Professor in the Department of English, Carnegie-Mellon University. For the past three years, he has directed the freshman composition program at Carnegie-Mellon. He has degrees in philosophy and communication and extensive graduate training in linguistics and computational linguistics. He has published dozens of articles on the teaching of reasoning and writing. He has also collaborated with John R. Hayes and Linda S. Flower on empirical studies investigating the writing process. In 1983, he received a seed grant from the Buhl foundation to evaluate current software aids for writers and to begin development on a new generation of software aids. The WARRANT project represents the chief design and development effort for this new generation of aids.
Christine M. Neuwirth is currently an instructor in the Department of English, Carnegie-Mellon University and Director of CMU's Writing Center. For the past three years, she has taught a graduate level course in "Computers and Rhetoric." She has degrees in computer science and philosophy. Her Ph.D thesis, which she is presently completing in the Department of English, is on the design and implementation of an intelligent tutor to guide students to write precisely under tight semantic constraints. She, like Kaufer, has carried out studies in the psychology of writing and reading and is well versed in conducting protocol studies. She has an extensive background in software engineering and the design of large systems. She is the principal designer of the WARRANT system and will oversee every phase of its implementation.

Cheryl A. Geisler is currently a Ph.D. candidate in the Department of English. She has a wide competence in empirical methods, computer programming, and reading and writing instruction. She has completed several empirical studies on writing evaluation and teachers' responses to student writing. She has conducted many protocol studies of students' writing and reading. She has written several computer simulations of aspects of the writing process. She has a masters degree in reading instruction and has taught reading to nonnative speakers. For the WARRANT project, she will coordinate the protocol studies of the novice and expert subjects in our model reading, reasoning, and writing tasks.

Ruth Von Blum, our primary consultant, is currently a member of the Department of Psychology at UCLA and—pending negotiations—will become the Associate Director of CDEC in the fall of 1984. She has a Ph.D in Science-Mathematics Education from the University of California, Berkeley. She is a nationally acclaimed educational technologist who designed WANDAH, widely considered the most advanced piece of writing software to date.

Robert H. Ramey, our chief programmer, is currently a senior scientist at the Computer Engineering Center at Mellon Institute. He has a masters degree in applied mathematics from Northwestern University. He has extensive experience working with ZOG, the state-of-the-art information management system we will be building into WARRANT. He also has a deep interest in education, having developed communication software and hardware for severely handicapped children.

John R. Hayes is a Professor in the Department of Psychology, Carnegie-Mellon University. He has degrees in physics and cognitive psychology from Harvard and M.I.T. He is a nationally recognized expert in the areas of human problem-solving and
creativity. For the past 10 years, his interests have taken him to the study of writing and reasoning skill. With Linda S. Flower, he pioneered protocol analysis as a method to tap the underlying processes of expert and novice writers. Along with Flower, his name is widely associated with introducing the "process" approach to the teaching of writing. He will serve as our primary consultant for our protocol studies.

7.3. VITAE of Project Personnel

[Vitae appended at end of proposal document]
The national literacy crisis has emerged from the confluence of two complementary, but quite distinct, social trends. The first is the apparent decline in writing skill among the college-age population. The literacy levels of high school students have sharply declined in the last twenty years. Increasing numbers of high school students with steadily declining verbal S.A.T. scores entered college in the 1960s and 1970s [Arthur Applebee, "The SAT-Score Decline Report" Support for Learning and Teaching of English Newsletter, 2 (November 1977), 1]. The second trend is the rise of the external demand for skilled writers in all walks of life. Though the literacy skills of college-bound students have seemed to decline, the demand for students with exemplary writing skill has skyrocketed.


That methods of normative reasoning and inquiry (in, say, philosophy or law) have general heuristic value for writers has been argued variously by: Carella in "Forensic Rhetoric and Invention: Composition Students as Attorneys," College Composition and Communication (February 1983); Covey in "Formal Logic and Philosophic Analysis," Teaching Philosophy (July/October 1981); Garver in "How to Develop Ideas: The Contribution Philosophy Can Make to Improve Literacy," Teaching Philosophy (April 1983); Hillocks in "Inquiry and the Composing Process," College English (November 1982); Kaufer in "A Plan for Teaching the Development of Original Policy Arguments," College Composition and Communication (February 1984); and Kaufer and Neuwirth in "Integrating Formal Logic and the New Rhetoric," College English (1983).


"Coherence, cohesion, and writing quality," College Composition and Communication,


21 KAPSTROM, INC. 5952 Royal Lane, Suite 124A, Dallas, TX 75230.

22 See Section 6.2.1, System Hardware, page 52.


29 Collette Daiute is currently studying inexperienced writers, in particular, children, using an easy to learn text-editor for composing. Her results are encouraging. See Collette Daiute, "Word Processing. Can It Make Even Good Writers Better?" *Electronic Learning* (March/April, 1982), 29-31.


33 The IBM Peaches have not yet been released. Carnegie-Mellon, under contract with IBM, expects delivery of the first of these machines in April of this year, with 100 machines available for faculty development efforts in the Fall of 1984.


35 The mouse has been found to conform to Fitt's law of practice, i.e., after some practice you can point with a mouse as easily and quickly as you can point with your finger. The limitations on pointing speed are those inherent in the human nervous system. See Stuart Card, William English and Betty Burr, "Evaluation of Mouse, Rate-Controlled Isometric Joystick, Step Keys, and Text Keys for Text Selection on a CRT," *Ergonomics*, 21:8 (1978), 601-613.

36 The display has 1000 by 800 dots resolution. In a bit-mapped display, every screen dot can be turned on or off by setting a corresponding bit in memory.


38 Carnegie-Mellon's Communication Design Center has developed considerable


40 Unix is a trademark of AT&T Bell Laboratories.

41 C is a trademark of AT&T Bell Laboratories.


45 The text-editor, Bank Street Writer, for example, was recently rated in *Consumer Reports* as easy to learn but relatively hard to use.


47 Since these machines will not be manufactured initially in quantity, their cost is even higher than that incurred solely by their increased capability.

48 We consulted on WARRANT's hardware requirements, feasibility of implementation, and timetable-budget with the following people: Don McCracken, Research Computer Scientist, Department of Computer Science, Carnegie-Mellon University; James Gosling, System Designer, Information Technology Center, Carnegie-Mellon University; and Robert Ramey, Senior Scientist, Engineering and Applied Sciences, Mellon Institute, Carnegie-Mellon University.

49 Raj Reddy, "Technologies for Learning," in *Computers in Education: Realizing the


51Neuwirth has done initial protocol research exploring how experts and novices differ in reading and composing argumentative refutations. See C.M. Neuwirth, "A Model for Finding Crucial Premises," Paper presented at the Conference on College Composition and Communication (San Francisco: March, 1982).


54Whether we allow students to access examples of writers pursuing those strategies before students have themselves attempted the strategy will depend on our observations of how students use the system. Anderson, Kulhavey and Andre studied students who were allowed to look ahead to a known correct response versus those who were not. When allowed to look ahead to a correct response, some students did not process the information as deeply as students who were not allowed to do so. See R.C. Anderson, R.W. Kulhavy and T. Andre, "Feedback Procedures in Programmed Instruction," Journal of Educational Psychology, 62:2 (1971).

55If feasible, we will structure this so that it is individualized for different students. In addition, we will structure this facility as a switch that can be turned on or off.


57Numerous preliminary studies are appearing on the effects of using text-editors on the process of composing. See, for example, the May, 1983 issue of College Composition and Communication as well as Bridwell, noted in Section 6.1, Our Design
Goals. However, none of these exploratory case studies control for the design of the text-editor.


61 For this reason, and because low-level editing aids are the focus of other development projects, we will not be implementing low-level style checking capabilities in WARRANT.
