Applying Human-Computer Interaction Methodologies to the Development of Computer-Aided Design Software

by

Jason Moo

Thesis submitted to the Faculty of the

Virginia Polytechnic Institute and State University

in partial fulfillment of the requirement for the degree of

Master of Science

in

Mechanical Engineering

APPROVED:

Jan Helge Bøhn, Chairman

Mary Beth Rosson

Arvid Myklebust

July 1995

Blacksburg, Virginia

Applying Human-Computer Interaction Methodologies to the Development of Computer-Aided Design Software

by

Jason Moo

Jan Helge Bøhn, Chairman

Mechanical Engineering

(ABSTRACT)

The scarcity of usability studies on computer-aided design (CAD) systems has prompted three questions: (1) Is usability testing different for CAD systems than conventional software? (2) What role does domain expertise play in the use of CAD software since most CAD applications are not simple tasks? (3) How does the design parameters' presentation affect the user's understanding of the design, the design domain, and the capabilities of the CAD system?

This thesis addresses these three questions through the application of human-computer interaction (HCI) techniques in the development of a graphical user interface (GUI) for the metal-ceramic brazing process: First, a GUI for the metal-ceramic brazing process was designed and implemented. Subjects were then invited to test the GUI by performing a set of benchmark tasks. The results of these tests were analyzed, both to characterize design flaws in the interface, and to suggest redesign solutions to observed problems. The GUI was then refined based upon the test results and retested.

To further aid in the understanding of the effects of the design parameters'

presentation, a usability evaluation was also performed for a finite element module of a state-of-the-art commercial CAD system (I-DEAS Master Series, version 1.3). This evaluation provided a comparison for the metal-ceramic interface with respect to interaction styles, presentation of domain knowledge, and task performance.

This study demonstrated that HCI methodologies can be applied effectively to CAD user interface development. In particular, it encouraged the development of a usability-oriented process for developing usable interfaces. However, it also highlighted the difficulties posed by the complexity of a CAD system and the associated task analysis.

It was observed that three distinct sets of knowledge are involved in the use of a CAD system: task knowledge, domain knowledge, and application knowledge. By separating and distinguishing the needs of these sets of knowledge, the CAD system developer is more likely to design a high-quality interface, one that supports the task performance while educating the novice user of the technical background of the application.

It was also found that a clearly and intuitively organized interface can provide an invaluable insight for users into the functionality of the system and facilitate users' learning of the system. The subjects who participated in this study were inexperienced in both the application and the application domain and relied greatly upon the organization of the interface to lead them through the benchmark tasks. The organization of the system functions or, in the case of the metal-ceramic brazing interface, the design parameters, was shown to be crucial to the subjects' understanding of the application because it was the only source of guidance the subjects had as to how to perform the benchmark tasks. In particular, it was observed that the subjects had the tendency to align their thought process to that of the interface in their attempts to understand the system.

Acknowledgements

I would like to thank my advisor, Dr. Jan Helge Bøhn, for his guidance and patience. I would also like to thanks the members of my Master's commitee, Dr. Mary Beth Rosson and Dr. Arvid Myklebust, for their time and invaluable insights.

Much love and thanks to my mother whose love and support always made the tough times easier. My love and thanks also go to Satomi whose encouragements and love kept me sane through the long nights in front of the computers.

Thanks to Dr. Deborah Sinnreich-Levi whose counsel and friendship was a steady source of support through the years.

Thanks to Ken, Karen, and Jun Cranker who treated me like a brother and provided a place for me to hide from my work.

Thanks to my roommates, David, Gorjun, Chen, and Yagna, for putting-up with me and being victims in my work.

Thanks to all the participants who took the time to help me in this thesis. Thanks for their patience and time.

I would also like to thank God for carrying my through this time. I look forward to the walks we'll have together in the future.

Table of Contents

List of Figures

List of Tables

CHAPTER 1. INTRODUCTION

1.1 Motivation

Advancements in analytic algorithms and graphical modelling have made today's computer-aided design (CAD) systems an extremely powerful tool for designers. However, these powerful tools are usually difficult to learn and to use. The ease and efficiency with which the user interacts with a system are considered, in the human-computer interaction (HCI) community, to be a measure of the quality of the user interface. Since a user sees and interacts with the CAD system through the user interface, it has the task of conveying the functionalities and procedures of the CAD system to the user. In large and complicated systems like CAD it is not enough for the user interface to just present the functionalities of the system; it should also present the functionalities such that the user can use the CAD system at an optimal performance level.

Although user interfaces have always been a part of CAD systems, they are typically developed without a definitive measure for the effectiveness of the interface or a paradigm for the style of interaction. Most computer users are familiar with the term *user friendly* and agree that it is of importance to the design of software packages. However, *user friendly* is a subjective, vague, and non-quantifiable term which cannot serve as anything but a jargon for interface design. HCI (a human factors discipline with the goal to optimize human-computer performance), however, is built upon empirical data, *usability,* and methods to improve the user interface. HCI opens a new pathway for addressing user interface development in CAD systems.

By considering the interface design from a user's point of view, HCI methods design the interface to address user tasks rather than system functionality. The system is designed to be a tool for the user to use rather than a collection of functions for the user to learn. HCI's usability addresses issues such as ease of learning, speed of user task performance, user error rate, subjective user satisfaction, and retention over time [Hix93]. These issues can be recorded, quantified, and analyzed to improve the quality of the user interface.

Formative evaluation is the empirical process which cyclically collects these usability data during the development of an user interface. Types of formative evaluation include [Hartson94]: (1) Subjective evaluation--user opinion, (2) Objective evaluation--directly observed user performance, (3) Qualitative evaluation--non-numeric data and results, (4) Quantitative evaluation--numeric data and results.

Formative evaluation is usually used in cooperation with a set of benchmark tasks, which are realistic user tasks such as draw a solid line from 0,0,0 to 1,2,0 in Professional CADAM, to measure specific usability attributes (e.g., initial performance, learnability, retainability, and first impression). The type, or types, of formative evaluation and the benchmark tasks used to evaluate an interface depends greatly on the domain of the system. Therefore, it is important to identify the users, the system mission, and the acceptable level of user performance in specific, quantifiable usability goals.

Metal-ceramic brazing has been chosen as the domain of the application because of its non-trivial task of design parameters refinement and the need to focus upon both the design parameters and the geometric model. The complex brazing rules and the set of twelve design parameters that are necessary to define a braze joint, makes metal-ceramic brazing a very specialized field that requires domain expertise on the part of the braze designer. The subjects chosen for this study had little or no knowledge about metal-ceramic brazing. This simulates quite accurately how most users initially approach a CAD system; they being novices both with respect to the application domain and the CAD system itself.

HCI practices can also address specific interaction problems such as visualization of data and presentation of non-geometric knowledge. Design parameters in CAD systems fall into this category and are in great need of HCI attention. The focus of the CAD community on graphical model representation and analytical methods has relegated informational and non-graphical components like design parameters to be handled by lists and menus on a data management basis. Menus and lists can be quite effective user interface techniques, but the large quantity of parameters that is involved in a CAD design makes navigation through lists and menus difficult. Lists and menus also hide these parameters, even though they are just as important to the design as the graphical model which usually dominates a CAD graphical user interface. Usability evaluation can bring to light which methods of parameter organization are most effective for a task, what information needs to be presented up front, and which can be placed into menus and lists.

CAD is a field that could greatly benefit from the infusion of HCI user interface design practices. Not only do more usable interfaces shorten training time and increase design efficiency (which means shorter production time and increase in productivity), they also have organizational implications. According to Robertson and Allen [Robertson92],

"... it is difficult for the user to gain proficiency in the [CAD] technology... CAD systems do not necessarily *cause* any changes to occur to the jobs, work process, or structure of an organization--they only enable changes... Many managers expressed frustration at the task of managing workers on CAD, a technology they did not understand." A more usable and more understandable CAD user interface would help organizations to make those changes they envisioned for their systems and give managers, who at best will be casual users of the systems, a better understanding of the work being done. The potential that CAD enables is largely unrealized by its users and HCI can be the approach to giving CAD users the powerful tool that they had foreseen.

1.2 Problem Statement

The marriage of HCI and CAD seems to be a likely match. However, a review of published literature has shown that no one has yet evaluated the usability of a CAD system in order to iteratively refine it. The lack of an usability evaluation study on CAD systems begs the following questions:

(1) Is usability testing different for CAD systems than conventional software?

(2) What role does domain expertise play in the use of CAD software since most CAD applications are not simple tasks?

(3) How does the design parameters' presentation affect the user's understanding of the design, the design domain, and the capability of the CAD system?

1.3 Solution Overview

The answers to the above questions were found by studying the effects of iterative refinement on an interactive graphical representation of the design parameters which helps the user perform design tasks and understand the design as well as the design domain. First, a GUI for the metal-ceramic brazing process was designed and implemented following well defined HCI user interface guidelines and techniques. Subjects were then invited to test the GUI by performing a set of benchmark tasks. The results of these tests were analyzed, both to characterize design flaws in the interface, and to suggest redesign solutions to observed problems. The GUI was then refined based upon the test results and retested.

To further aid in the understanding of the effects of design parameters' presentation, a usability evaluation was also performed for a finite element module of a commercial CAD system, specifically, I-DEAS Master Series, version 1.3. I-DEAS was chosen because it is recognized as one of the best user interface in their industry, especially after recent redesigns. This evaluation provided a comparison for the metal-ceramic interface with respect to interaction styles, presentation of domain knowledge, and task performance as well as a control comparison with the state-of-the-art in CAD user interface.

1.4 Organization of the Thesis

Chapter 2 presents the background of the three disciplines that were involved in this thesis: CAD, HCI, and metal-ceramic brazing.

Chapter 3 is a literature review of the three disciplines mentioned above along with observations on the literature reviewed and how they applied to this thesis.

Chapter 4 presents the work that was completed, detailing the initial and the redesigned metal-ceramic user interface as well as the formative evaluation that was performed.

Chapter 5 presents the qualitative data collected from the formative evaluation on the I-DEAS finite-element module and the initial and the redesigned metal-ceramic user interfaces.

Chapter 6 presents an analysis of all the qualitative data as they pertained to the three questioned posed by this thesis in Chapter 1.

Chapter 7 presents the contribution that were made from this work and directions for future work.

Chapter 8 is the list of references used in this thesis.

CHAPTER 2. THESIS BACKGROUND

2.1 Computer-Aided Design

Computer-aided design (CAD) can refer to any kind of activity which utilizes a computer to assist in the creation, modification, presentation, and analysis of a design. In any design process there are three major functions: synthesis, analysis, and presentation. The synthesis stage of design includes mental processes such as decision making and creativity. The synthesized objects are represented in a generally accepted format, such as an engineering drawing or a physical model. After synthesis, the object is then analyzed, evaluated, and modified. After several iterations, the final design can be prepared for presentation and approval. In a traditional design process or non-CAD system, all three functions are carried out manually with the aid of drafting instruments, calculators, handbooks, tables, charts, etc. A computer-aided system uses a computer and input/output (I/O) devices to substitute for these traditional aids.

The term CAD is also used to refer to any of the sub-functions of the computer-aided design system; for instance, the design drafting program, the finite-element analysis program, or the structure analysis program. In fact, CAD has also been associated with only the utilization of interactive computer graphics with design functions. Hence, although some design functions do not necessarily make use of interactive computer graphics, it is generally accepted that the graphical user interface is an integral part of a CAD system [Majchrzak87].

2.1.1 A Brief History of CAD

The development of CAD can be dated back to the early 1960's when researchers at the Massachusetts Institute of Technology (MIT) developed the Sage system, in which they introduced the CRT display. The pioneering work of Ivan Sutherland on developing the Sketchpad interactive computer graphics system at MIT enabled the use of the CRT system as an electronic drafting board [Majchrzak87]. In the early 1960's there were also systems using APT-like (APT is a part programming language used in the control of numerical control machine tools) commands to control a plotter to draw two-dimensional engineering diagrams.

Two-dimensional drafting, three-dimensional drafting, wireframe CAD systems, and sculptured surfaces [Bezier72] were all developed in the 1960's shortly after interactive computer graphics devices became available. However, in these early years, due to the high cost of computers and peripheral devices, CAD was considered a luxury. In the 1970's, with the drastic drop in computer software and hardware costs, increases in their capability, and the development of user-friendly software, CAD became more widely accepted.

During the 1980's, software for 3-D solid modelling and shaded color graphics were developed. Many software programs for engineering analysis applications were also integrated into CAD systems. At this time CAD was moving from computer-aided drafting to computer-aided design and modelling. Developmental work in CAD moved toward interfacing standards and powerful and specialized hardware and software. The development of super-minicomputers in the late 1970's and super-microcomputers in the early 1980's added new dimensions such as solid modeling and expert systems to CAD. Now in the 1990's, computer-aided design, engineering, and manufacturing system is a multi-billion market and CAD has become an affordable tool for nearly every company.

2.1.2 Finite-Element Method

Finite-element method (FEM) essentially refers to a process through which a "continuum with infinite degrees of freedom can be approximated by an assemblage of subregions (or elements) each with a specific but now finite number of unknowns. Further, each such element interconnects with others in a way familiar to engineers dealing with discrete structural or electrical assemblies" [Zienkiewicz71]. Finite-element analysis (FEA), on the other hand, refers to the application of FEM to derive an approximation of, for instance, stresses. FEM has been understood, at least in principle, for more than 50 years [Fenner75]. The integral formulation on which it is based has been known even longer. However, the method could not be applied in a practical way until the advent of the modern computer since it involved the solution of a large number of algebraic equations. Today, these thousands of non-linear algebraic equations that make up an FEA problem can frequently be solved within minutes.

Mechanical engineers, confronted with complicated structural problems in aircraft and ship design, were the first to take advantage of the advanced computational methods and high-level languages to transform mechanical models into algebraic equations. FEM has been studied by applied mathematicians and has become a part of the more general study of partial differential equations. Today, FEM has become a popular approach to solving complicated problems for all engineering disciplines because of the relative ease with which it can be applied to geometrically complicated systems [Fenner75].

2.1.3 The Basics of FEM

All finite-element methods involve dividing the physical system, such as a cantilever beam, into small regions or elements. The structure that has been divided into elements looks like a grid or a mesh, hence, the process of dividing the structure into elements is called mesh generation. Each element is an essentially simple unit, the behavior of which can be readily analyzed. The complexities of the overall systems are accommodated by using large numbers of elements rather than by resorting to the sophisticated mathematics required by many analytical solutions. It is generally a good practice to avoid using a mesh with irregularly shaped elements. However, higher degree elements have been developed such that these irregularities can be better approximated.

The basic steps of FEM for stress analysis are: (1) Break the structure into elements. (2) Find the stiffness of each element. (3) Assemble the element stiffnesses into a matrix which defines the stiffness of the whole structure. (4) For the given load and restraints, use the stiffness matrix to determine the displacements of the nodes of the elements. (5) Using the displacements, calculate the stress in the elements, and thereby the stress in the structure [Bowes75].

2.1.4 General Architecture of FEA CAD Systems

In practice, all FEA CAD systems have three distinct modules: (1) the data entry module (preprocessor), (2) the solver, and (3) the postprocessor. The data entry module is used for entering all the information necessary for the analysis of the problem by the FEM. This includes the description of the geometry of the structure, the meshing of the structure, and the description of the physical characteristics of the structure. The latter includes material characteristics, loads, and constraints. The solver computes the unknowns, e.g. displacements and stresses, in the finite-element problem. Finally, the postprocessor extracts the information computed by the solver and presents the numeric information in a more understandable graphical form such as using color shading to illustrate stress across the structure. The finite-element module of I-DEAS Master Series, which this thesis made use of, follows this general architecture.

2.2 Human-Computer Interaction

Human-computer interaction, often referred to as HCI, is what happens when a human user and a computer system get together to perform tasks. The study of human-computer interaction is a field of endeavor devoted to answering the question of how best to make this interaction work. As a field, human-computer interaction includes user interface hardware and software, user and system modeling, cognitive and behavioral science, human factors, empirical studies, methodologies, techniques, and tools. The goal of most works in human-computer interaction is, in one way or another, to provide the user with a high degree of usability [Hix93].

The following sections on HCI presents some basic concepts in HCI such as usability, human factors information, and user-centered design. There are also sections which explain formative evaluation as well as detailing the different elements that makes up a formative evaluation. Tools which aid in the development of user interfaces such as tool kits and user interface management systems (UIMS) are also reviewed.

2.2.1 Usability vs. User Friendliness

Most computer users are familiar with the term *user friendly* and agree that it is of importance to the design of software packages. However, user friendly is a subjective, vague, and non-quantifiable term which cannot serve as anything but a jargon for interface design. HCI, however, is built upon empirical data, *usability*, and methods to improve the user interface. Usability is comprised of ease of learning, speed of user task performance, user error rate, subjective user satisfaction, and retention over time. Usability is related to the effectiveness and efficiency of the user interface and to the user's reaction to that interface. The intuitiveness of the interface for the user is also an important aspect of usability [Hix93].

2.2.2 Human Factors Information

There are several kinds of human factors information that pertains to computer interface design: standards, design guidelines, and style guides, both commercial and customized [Hartson94]. Standards are official, publicly available documents that state, in general wording, the requirements for a design. Standards are usually enforced on a design either by contract or by law; e.g., "Standardization. The content of displays within a system *shall* be presented in a consistent, standardized manner." Guidelines give design guidance in the development of an interface, and are usually publically available and sometimes are empirically derived and/or validated; e.g., "Consistent format. Adopt a consistent organization for the location of various display features from one display to another. For example, one location might be used consistently for a display title, another area might be reserved for data output."

Commercial style guides are usually produced by an organization to define a specific, certifiable interface style including both *look* (appearance) and *feel* (behavior); e.g., "Pop-up menus save space. Pop-up menus have the advantage that they take up no permanent screen space. Not being associated with a menu bar, they simply pop up at the current pointer location." Customized style guides are specific design requirements that are internally enforced within an organization; e.g., "Display date and time, in MM/DD/YY HH:MM AM/PM format (e.g., 4/1/94 10:13 PM) in a small box in the upper right-hand corner of each screen" [Hartson94].

One of the biggest concerns about the various kinds of human factors information is the possibility that they can be misused. By their very nature, they tend to be too vague (standards and guidelines) or too specific (commercial and customized style guidelines). Perhaps the greatest pitfall in using these kinds of human factors information is that developers will use them as an excuse for no longer doing usability evaluation with human users. Standards, design guidelines, and style guides may contain a lot of good human factors information, but that information alone is not enough. The process by which that information is used, and the way in which the resulting interfaces are evaluated, constitutes a major portion of the effort involved in producing a high-quality user interface [Hix93].

2.2.3 User-Centered Design

User-centered design addresses the design of systems, computers, and interfaces from the point of view of the user. Its focus is on people, rather than on technology. It is concerned with the goals and needs of the users, the tasks they wish to perform, and the methods they prefer for performing these tasks [Norman86]. Its goal is to produce systems that support the work people actually do. Unfortunately, what is best for the user is rarely the easiest for the interface designer or programmer to implement. User-centered design takes time, effort, and expertise. Producing effective user interaction requires focusing on what is best for the user rather than what is quickest and easiest to implement.

User-centered design involves incremental steps with users to provide them with progressive mastery of the evolving system. This approach is implicit in the recommended principles of interface design [Gould85]: (1) Early and continual focus on users and tasks. (2) Empirical evaluation with real users and prototypes. (3) Iterative refinement (i.e., design, test, and measure) throughout development.

User-centered design is embodied by the maxim *know the user* which means to *understand* (not just identify, describe, or stereotype) the users of the system through techniques such as user and task analysis. To know the user also means to prevent user errors. This means to anticipate possible user errors and to design a system that will not allow such errors. A well designed interface leaves almost no user action unanticipated.

Another vital part of user-centered design is that the user feels in charge at all times, rather than feeling the computer is in charge. The user should have the impression that the computer is prepared to respond to whatever the user is ready to do.

2.2.4 Formative Evaluation

Formative evaluation is the evaluation of the interface design as it is being developed, both early and continually throughout the interface development to improve the interface. In contrast, summative evaluation occurs after a system design is complete, often to compare existing systems. Formative evaluation produces quantitative data against which developers can compare the established usability specifications, and it produces qualitative data that can be used to determine how to improve the usability of the interface design.

2.2.5 Types of Formative Evaluation Data

There are four types of formative evaluation data: (1) Subjective--These represent opinions, usually of the user, concerning usability of the interface. This kind of data is useful in gauging the level of user satisfaction and the intuitive learnability of the interface. (2) Objective--These are directly observed measures, typically of users' performance while using the interface to perform benchmark tasks. (3) Qualitative--These are non-numeric data and results, such as lists of problems users had while using the interface and suggestions for modifications to improve the interaction design. This kind of data is useful in identifying which design features are associated with measured usability problems during all cycles of iterative development. (4) Quantitative--These are numeric data and results, such as user performance metrics or opinion ratings. This kind of data is key in helping to monitor convergence toward usability specifications during all cycles of iterative development [Hartson94].

The subjective, objective, and qualitative formative evaluation data were the main focus of this thesis for three reasons: (1) It is difficult to set intelligent quantitative usability specifications to show convergence without previous formative evaluation on FEA and metal-ceramic brazing user interfaces. (2) The purpose of this thesis was not to develop a commercial product but to introduce HCI methodologies to CAD software development, therefore only one iteration was performed. With only two data points, i.e., the usability specifications for the initial and the redesigned metal-ceramic brazing interface, the convergence or divergence can at best only be indicated and were therefore not considered. (3) Their usefulness in measuring the intuitive learnability and in focusing on the cause of usability problems is well suited for the purpose of this thesis.

2.2.6 Formative Evaluation Steps

The major steps in formative evaluation are: (1) developing the experiment, (2) directing the evaluation sessions, (3) collecting the data, and (4) analyzing the data, drawing conclusions to form a resolution for each design problem, and redesigning and implementing the revised interface.

2.2.6.1 Developing the Experiment

Developing the experiment includes selecting participants to perform tasks, developing benchmark tasks for participants to perform, determining protocol and procedures for the evaluation sessions, and pilot testing to shake down the experiment. It is extremely important that the participants selected for the experiment represent the typical users who are expected to use the interface being evaluated. It is also important that the benchmark tasks simulate real world tasks that are expect of the system.

The evaluation session may include different protocols or procedures depending upon the types of formative evaluation data and the types of benchmark tasks that are used. However, a set of introductory remarks should be prepared and given or read to the participants at the beginning of the experiment. These remarks should briefly explain the purpose of the experiment, tell a little bit about the interface, state what the participant will be expected to do, and the procedure to be followed by the participant. It is also important to specifically make clear to all participants that the purpose of the session is to evaluate the system--not them--since some participants may be fearful that their performance during the session may reflect poorly on them. In this regard, it is effective to guarantee the confidentiality of individual information and anonymity of the data.

Another important part of the test session is the preparation of the informed consent form which must be signed by participant before the start of each test session. This form states that the participant is volunteering for the experiment, that the data may be used if the participant's name or identity is not associated with those data, that the participant understands the experiment is in no way harmful, and that the participant may discontinue the experiment at any time. This is a standard protocol for performing experiments using human participants, and protects both the evaluator and the participant.

The informed consent form is legally and ethically required; it is not optional.

2.2.6.2 Directing the Evaluation Session

Typically the experimenter will be in the same room as the participant during a test session. The exception is for quantitative measures of performance, in which case the evaluator should be in the background or choose to be next door in a control room so as not to be obtrusive.

First the experimenter should briefly explain the lab set up and settle next to the participant. The informed consent form should then be read and signed by the participant after which, the introductory remarks should be administered. Next, the participant can be given the task instructions and begin performing the benchmark tasks.

During the session, the participant may be asked to explain the rationale behind selections or idleness especially during qualitative data collection to obtain the desired information. Questions such as "What are you looking for?", "What are you trying to do?", or "What did you expect to happen when you clicked here?" often are helpful to understand the participant's understanding of the interface.

The experimenter should refrain from commenting or giving help during the session. However, to encourage the participant to complete the tasks, the experimenter may choose to hint the participant as to how to complete the task with which the participant may be struggling. However, the experimenter should not give a participant specific instruction as to how the task can be completed. By telling a participant the actions to perform, the evaluation looses the information that would otherwise be acquired as the participant continues to try to complete the task.

2.2.6.3 Qualitative Data Generation Techniques

Qualitative data are sometimes more elusive than quantitative data, however, qualitative data are extremely important in evaluating the usability of an interface. The kinds of techniques that are most effective in generating qualitative data include: concurrent verbal protocol taking, retrospective verbal protocol taking, critical incident taking, and structured interviews.

Verbal protocol taking, sometimes called "thinking aloud," [Ericsson84] is perhaps the most common technique for qualitative data generation. This approach is immensely effective in determining what problems the test subjects are having and what might be done to alleviate those problems. In concurrent verbal protocol taking, the experimenter asks the participants to talk aloud while working through an evaluation session, indicating what they are trying to do, why they are having a problem, what they expected to happen that didn't, and what they wished had happened. This is in contrast to retrospective verbal protocol taking, where the participants perform the benchmark tasks relatively uninterrupted during a taped session. Then, immediately after the session, the experimenter and the participant review the videotape together and analyze the session. Because concurrent verbal protocol taking is invasive and may not be natural for all participants, the retrospective verbal protocol taking may be utilized instead. However, the biggest draw back to retrospective verbal protocol taking is time. The concurrent verbal protocol taking method was utilized for this thesis' evaluation sessions.

A critical incident is something that happens while a participant is working that has a significant effect, either positive or negative, on task performance or user satisfaction. A negative critical incident is typically a problem a participant encounters--something that causes an error, blocks progress in task performance, or results in a derogatory remark by the participant. An occurrence that causes a participant to express satisfaction is a positive critical incident. While negative critical incidents indicate problems in the interface design, positive critical incidents indicate interface designs which work well. Critical incidents can be observed during performance of benchmark tasks, or when a participant is freely using the system.

Structured interviews are typically conducted after the experiment in the form of a series of preplanned questions that the evaluator asks each participant. The questions may include general questions such as "What did you like best about the interface?" or more detailed questions such as "What did you think that icon meant?" Structured interviews provide another form of qualitative data, but with the danger of producing invalid or unreliable data if the interview is poorly structured.

2.2.6.4 Analyzing the Data and Redesign

The major decision after the test sessions is whether to accept the interface design as it is or consider a redesign. This decision must be made at a global level, considering the interface metaphor, as well as at a detailed level, considering the individual problems. To make this decision, the interface designer may consider (1) the quantitative data in comparison with the preestablished usability specifications, (2) problems such as interaction design flaws or user difficulties directly associated with an interaction design flaw, (3) the effect on user performance as indicated by the amount of time a user spends dealing with a specific problem, (4) the importance of an observed problem on user performance, interface usability, and overall system integrity, (5) the solutions to each of the observed problems, and (6) the cost of making the changes indicated for each proposed solution.

Usually, the deciding factor in implementing a change in the existing design is the cost-importance analysis. In a cost-importance analysis, the interface problems are ranked with respect to their impact on system performance and their cost of elimination. The implementation problems should be ranked in the order of importance from high to low. If the problems are of equal importance, then the problem with the lower cost solution should be ranked first. Rankings are then usually divided into three categories with respect to implementation: one for those problems that definitely are going to be addressed, one for those to be addressed if there is time, and one for those that are postponed for now.

2.2.7 Toolkits

A user interface toolkit is a library of call routines, used by the programmers, for implementing low-level interface features. Toolkits provide code for various kinds of interaction techniques such as push buttons and number boxes. The function of toolkits is very limited when compared to other interface development tools. However, they are very flexible in the sense of programming. They provide no support for any activity in the user interface development process except implementation.

The metal-ceramic brazing interface developed in this thesis was implemented using a Motif-like toolkit developed by Woyak [Woyak92, Woyak93].

2.2.8 User Interface Management Systems

User interface management systems (UIMS) are user interface development tools that provide support for building consistent presentation and interaction components of a user interface to an application. UIMS mediate between the user and the application while hiding most of the application details from the user. First-time users may find UIMS' support functionality ideal in aiding the rapid adoption of state-of-the-art interface methodologies. UIMS may also be customizable, allowing the more advanced users to create, modify, and add new interactions techniques to an existing library.

The general architecture of a UIMS is comprised of (1) a graphical user interface presentation component which controls the graphical part of a user interface and is dependent on software subtools such as an underlying graphical system, (2) a fully integrated dialogue development environment which generally consists of a high level interaction dialogue specification language and its compiler, and (3) an application interface module which specifies the application specific data structure and the application procedures available to the user interface through callbacks and constraints. A UIMS should also provide good run-time support for the executing application. Together, these components support a variety of interface development tasks such as rapid prototyping, testing, modification of initial design, and evaluation.

The character of a UIMS depends on the level of support its components offer non-programmer interface designers. The principle characteristics of the interface development that should be supported by a UIMS are [Sastry92]: (1) All aspects of an application's interface such as its layout design and dialog specification can be developed as separate reusable modules. (2) The application is totally separated from its user interface. This allows the independent development of both modules, possibly by an application programmer and a non-programmer interface designer respectively. (3) It is possible to test and modify the user interface without any application code. This functionality eliminates the need to recompile the complete application during the iterative interface development cycle. (4) There is effective support for group project development.

Unlike toolkits, UIMS are interactive systems whose end-users are application programmers and/or interface designers. Hence the focus is on the system to provide a transparent and well structured interface to itself. The UIMS should aid the learning process of the interface developer as well as positively influence the quality of interfaces developed using it. A UIMS such as Graphic Modelling System (GMS), TeleUSE, or USEIT would have been the better choice of interface development tool for this thesis. However, such a system was not available at the CAD Laboratory at Virginia Tech.

2.3 Metal-Ceramic Brazing

Metal-ceramic brazing is an engineering process which bonds a metallic and a ceramic part, at the molecular level, usually by melting a brazing material between the parts.

Since its beginnings in the late 1930's, the technology of joining ceramics to metals has progressed steadily. Because of the promise of ceramics, it is a highly sought after by the automotive and aerospace industry. However, metal-ceramic brazing is a non-trivial process which may involve iterative adjustments to many of its design and process parameters. These adjustments depend largely on the parts' geometry, their material properties, and other parameters controlling the brazing process. Most of these design parameters are interdependent, therefore, modifications to one of these parameters may necessitate changes to others. The process, initially more of an art than a science, through years of development, refinement, and improvement of material purity, processing, equipment, and techniques, has become an industrial production process that is based on a sound understanding of the principles, mechanisms, and reactions which occur during all phases of the ceramic preparation and subsequent cleaning, coating, and the prepping of surfaces, and the consolidating, joining, postcleaning, and heat treatment [Schwartz90].

The main area of concern in the metal-ceramic brazing process is the braze filler and its compatibility with the ceramic and metal parts because of the stresses produced in the joint between the metal and ceramic parts by the thermal expansion difference. Numerous experimental and theoretical results have been collected to generated a database which matches braze fillers and their proper application in metal ceramic brazing. Still, braze design is complicated by the stresses in the joint being dependent on the geometry of the joint and the differing coefficient of thermal expansion (CTE). Designers, therefore, not only need to be mindful of the CTE differences between the parts, but also of the geometry changes that are due to the temperature changes that occurs during the brazing process.

A set of twelve design parameters have been identified by Bass [Bass95] to be the primary contributors in a metal-ceramic braze design. They are: metal material, ceramic material, filler material, cost, operating temperature, braze strength, braze stress, braze technique, braze atmosphere, braze heat schedule, gap size, and the geometry of the braze surfaces. These twelve parameters formed the basis of the metal-ceramic brazing interface for this thesis.

CHAPTER 3. LITERATURE REVIEW

3.1 HCI and CAD

Human-computer interaction and computer-aided design, both being well established disciplines, have produced and continue to produce an abundance of literature. The literature presented here is representative of the literature that is relevant to this thesis, namely the application of HCI to CAD user interface development.

3.1.1 User Interface: Definition and Importance

It is important for the user interface designer to obtain a clear definition of what constitutes a user interface before undertaking the task of designing an interface. It is also important to understand the importance of the interface and the interface's role in a computer package.

Bass and Coutaz [Bass92] defined the user interface as the mediator "between two main participants: the operator of the interactive system (a human being) and the computer hardware and software that implement the interactive system." Robinson [Robinson90] noted that "the interface stands as the boundary between what is the work of HCI and what is the work of the software engineer. The 'true' nature of the system - what it does - is defined by the state of the software 'behind' the interface... the neutral and objective functionality of the 'internal' state (of the machine) may be realized by alternative interface designs." Hix and Hartson [Hix93] stated "to the user, the interface *is* the system."

A study by Myers and Rosson approximated "49% to 99% of interactive system code supports the human-computer interface" [Myers92]. Zecher [Zecher94] wrote: "user interfaces have become so important to most users that it is often a primary factor in determining which software package a user ends up purchasing." Hicks and Essinger [Hicks91] considered computers as the ultimate tool and pointed out that the definition of a tool emphasizes a human-centricity, "that is so obvious, no-one usually notices it."

3.1.2 User Interface Design Approaches

A user interface design approach helps the interface designer follow a systematic methodology in designing the user interface. The correct approach can help the designer to see possible design problems and address them before implementation.

Brown and Cunningham [Brown89] presented an approach to the user interface design process from a programmer's point of view through generally accepted interaction styles for programmers to follow along with guidelines for programmer to expand on their interface. Hutt [Hutt93] dissected the user interface from a business' stand point. Subjects such as trends in user interface, legal obligations, and user interface qualities were listed and briefly defined.

Bass and Coutaz [Bass92] recommended six phases for computer systems' design and development: (1) *Requirements definition* is the formulation of a formal or semi-formal statement of the problem to be solved. (2) *Specification* consists of high-level functional design and internal design. (3) *Implementation* expresses the internal specification in terms of a set of programming languages and tools. (4) *Testing* involves debugging both modules by themselves and as sets of modules. (5) *Installation* places the software system into production. (6) *Maintenance* involves making changes and dealing with their side effects. Bass and Coutaz also found that the user interface life cycle generally follows the six phases of software engineering and added two observations in regard to the user interface design process: (1) Designing and developing the user interface is an iterative process of refinement; thus, an implementation should be based on a software architecture that supports modification of the user interface. (2) The user interface contains both a human side, which is studied by psychologists, and a computer side, which is described by a hierarchy of machine abstractions. These abstractions provide both various levels of services and differing costs due to implementation and maintenance. The user interface designer must therefore be aware of both components.

Hancock and Chignell [Hancock89] described four approaches to the design of user interfaces: (1) The empirical approach uses an experimental tool to tease out the features of the interface that are easy or difficult to use. (2) The predictive modeling approach tries to predict the best design in terms of human-computer interaction performance before it is prototyped. (3) The anthropomorphic approach is the application of human qualities to the user interface to mimic human-human communication. (4) The cognitive approach applies the understanding of how human perceive, store, and retrieve information in designing the user interface.

3.1.3 HCI User Interface Guidelines

HCI user interface guidelines are rules of thumb that have been accumulated over years of user interface design by the HCI community. They serve as general guides and warnings for pitfall in user interface design.

Halter [Halter85] presented a checklist of general do's to designing a good user interface: keep it simple, know the operator, include plenty of feedback, inform operators of errors, tell operator how to recover, make errors easy to correct, avoid flashing lights and sirens, don't lull operator into false security, and build according to the user's level and needs. In addition to the checklist, Halter also stressed that the most important interface design rule is to make it easy to use, to learn, and to remember.

Aaron [Aaron84] defined the seven characteristics he considered to be the foundation of a good interface: (1) Simplicity--major parts of the system should be few in number or should be hierarchically organized, (2) Clarity--the parts of the system should be evident, (3) Familiarity--the parts of the system should remind user of things already known, (4) Integrity--the system should be an ordered sum of its parts, (5) Consistency--what the user knows of one part should help with other parts of the system, (6) Reliability--the system should respond to user in a trustworthy manner, and (7) Responsiveness--the interactive replies of the system should be quick, polite and helpful.

Foley et al. [Foley84] stated that "the quality of an interaction design is determined by some combination of the following primary criteria: (1) the <u>time</u> any user must spend accomplishing a particular project which the system is intended to support, (2) the <u>accuracy</u> with which the user can accomplish the project, and (3) the <u>pleasure</u> the user derives from the process." A set of secondary criteria which supplements the primary criteria was also presented: learning time, recall time, short-term memory load, long-term memory load, error susceptibility, fatigue susceptibility, naturalness, and boundedness.

Schneiderman [Schneiderman88] contributed the "three pillars" of support for user interface design process: guideline documents, UIMS, and usability laboratories for iterative testing. He also identified five primary interaction styles: menu selection, form fill-in, command language, natural language, and direct manipulation.

Spencer [Spencer85] focused on the importance of human memory limitation, visual reaction, menu organization in the interface design, and how these points should be considered from the user's perspective instead of the system designer's perspective. Swezey and Davis [Swezey83] also worked on the human factors of human-computer interaction, but most of their discussion revolved around issues of ergonomics such as resolution of graphical displays, brightness, flicker, keyboard, and workspace.

Hix [Hix93] and Hartson [Hartson94] presented a comprehensive set of general interface guidelines including: user action notation, formative evaluation, iterative refinement of interfaces and a working definition of usability. Intended as course text and notes for Computer Science 5714, Virginia Polytechnic Institute and State University, Fall 1994, these guidelines present a systematic approach to user interface design and its subsequent refinement.

Powrie [Powrie87] pointed out that "most decisions have been taken on an ad hoc basis relying on intuition and, where available, research data and guidelines. Usable data has been lacking in many areas... Guidelines have been helpful as checklists of matters for consideration, but over-general or too context-dependent to be useful as exact models."

3.1.4 User Interface Management Systems and User Interface Design Tools

As Wilson and Conway [Wilson91] stated, a good user interface is very difficult to create. The purpose of user interface management systems (UIMS) and various user interface design tools is to take some of the burden off the interface designer and allow him or her more latitude in experimenting new design ideas and better pin-point design flaws.

Fischer [Fischer89] discussed the establishment of evaluation laboratories and methods by developers and the integration of user interface technologies into new application tasks. He argued the importance of presenting human factors design guidelines to designers and the establishment of evaluation laboratories. Lee [Lee90] and Myers [Myers89] described UIMS as interactive prototyping tools for developing user interfaces. These too need to be designed according to HCI practices.

Pittman and Kitrick [Pittman90] constructed an object-oriented UIMS which consists of "a collection of objects and a semantically rich token language" that supports reconfiguration of components' presentation and interaction. Douglas et al. [Douglas90] constructed an abstract-object-oriented UIMS which allows the user to manipulate each object in its four components: graphic representation, functionality, spatiality, and attributes. Dewan [Dewan90] described Suite, developed at Purdue University, which supports: "(1) an input model that provides users and programmers flexibility of choosing when a particular kind of feedback is given in response to the modification to an active value, (2) an inheritance model supporting both IS-A and IS-PART-OF inheritance, (3) loose physical coupling between an application and its UIMS, and (4) integration of user interface software with operating systems, distributed systems, and database programming languages." Alty and Mullin [Alty89] demonstrated the dialogue specification technique in GRADIENT which separates out the functionality of an interaction into an "Application Interface Model, a Dialogue Control Component and a Presentation Component." The technique was tested in a real application and was found to provide considerable benefits.

Bailey [Bailey93] concluded that iterative refinement does improve usability, however the user interface designers "need to learn how to fix problems without introducing new problems." Bailey also stated the need for new tools and methods that allow designers to easily and quickly design and test radical change ideas, e.g., those that require prototyping from scratch and which hence are currently too expensive to pursue.

The metal-ceramic brazing interface of this thesis was implemented using a collection of Motive-like widgets created by Woyak [Woyak92, Woyak93]. This collection offers some basic interface widgets like push buttons and number boxes. The widgets were created as objects in C++ using graPHIGS.

3.1.5 Intelligent Interface Research

Intelligent interface research tries to do for user interfaces what artificial intelligence (AI) did for expert systems. Incorporating artificial intelligence and HCI, intelligent interface research is based upon four propositions: (1) Serious interface problems are ultimately semantic problems. (2) These semantic problems cannot be solved through good interface technique alone. (3) These problems cannot be solved by AI alone. (4) What is needed to address these problems is a synthesis of the two perspectives [Miller91].

Intelligent interface research seeks to provide solutions for large, complex problems that challenge solution through algorithmic techniques.

Intelligent interface research appears to be well suited for developing user interfaces for CAD systems. However, this is a relatively new approach and no articles were found that addresses its application in CAD systems.

3.1.6 Task-Action Grammar

According to Payne and Green [Payne86] "The central aim of task-action grammar (TAG) is to formalize [the mapping from the task level to the action level] in such a way that simple metrics over the grammar, such as the number of rules, will predict aspects of the psychological complexity of the mapping... The secondary aim of TAG is to help the analyst appreciate the structure of a task language." Booth [Booth91], Grant and Mayes [Grant91], and Schiele and Green [Schiele90] discussed the notion of consistency concerning TAG. They agreed that TAG provides a model into understanding the user's mental representation of specific properties of an interface as well as being a vehicle to analyze user consistency.

Although this thesis did not make use of TAG to study the structure of the CAD task language, it did draw upon the idea of TAG to understand how user tasks are performed in a CAD system. For example, the task of performing a stress analysis using finite-elements on a CAD system can be, and often is, presented in a manner that is different from the presentation in finite element stress analysis literature. The idea of TAG was utilized in this thesis as a reminder of what is required to perform the task, both by the task itself and the CAD system. This helped in understanding why even domain experts might struggle with a CAD system targeting their field of expertise.

3.1.7 Visualization

How best to present information graphically has long been an important issue in HCI. The main concern in visualization is how to represent information and concepts such that they can be easily seen and understood by the user. This is also a major topic of concern in the design of CAD packages because of the engineering need for large quantities of accurate data and graphical representation of designs.

Woodgate [Woodgate85] studied the way people learned to use computer programs and observed that video games based upon direct visual stimuli required no instruction and enjoyed a high acceptance level over a broad spectrum of users. However, the study also found that a large number of different icons were required for a simple spreadsheet application and that it was difficult to design so many unique and meaningful symbols which the users can quickly associate to its respective functionality. Meyer [Meyer92] argued that current metaphors are not sufficient in the context of spatial information systems and proposed a new metaphor, the blackboard metaphor, which takes advantage of the naturally intuitive properties of spatial data by allowing the user to converse with the database through spatial sketches.

Furnas and Zacks [Furnas94] introduced the multitree structure for representing information in a natural and hierarchical context. Furnas pointed out that the trees within a multitree afford a familiar notion of content and context for the users. Brown and

Sarkar [Brown93] presented a study in the use of a fisheye view on a graph which represented direct routes between major cities in the United States. The fisheye view proved to be an useful technique to view both local details as well as global context of how that area fits into the entire structure. Rao and Card [Rao94] presented a similar technique to view tabular information. Ahlberg and Schneiderman [Ahlberg94] devised the starfield display to allow the user to browse and iteratively refine search parameters and filter the information set. Kiem and Lum [Kiem92] described the GRAphical Database Interface (GRADI) of a multimedia database management system which supports the human query specification process such as incremental query specification and predefined joins.

Wu et al. [Wu92] presented a hybrid data structure which combines all of the design data needed throughout multiple levels of abstraction including behavior, structure, and floorplan into a single unified view. Magic Lens filters are another new user interface tool that combines an arbitrarily shaped region with an operator that changes the view of objects viewed through that region. Such tools can provide various types of information, ranging from text documentation to scientific visualizations, and are advantageous in their physical lenses metaphor, by viewing objects in context, and by limiting clutter [Stone94]. Schleich and Durst [Schleich94] attributed the difficulty of graphics editors to the fact that only a small part of the information is visible and therefore accessible and understandable. They presented a visualization in the form of trees and graphs in a structure browser which displays the hierarchy of primitives in a graphic object.

Visualization is of special interest to this thesis because of the need to interactively present non-geometrical design parameters. Unlike informational data, the design parameters in metal-ceramic brazing not only convey design information to the designer but must be iteratively adjusted through a complex set of brazing rules. Because the designer may or may not choose to personally adjust the parameters, depending upon the design requirements, the presentation of the design parameters must be both informative and easily accessible for manipulation. Furthermore, since metal-ceramic brazing is a specialized engineering field, the interface must also support users with minimal domain expertise by guiding them in the process of metal-ceramic braze design.

3.1.8 CAD Usage Environment

The environment under which CAD systems are being used in corporations, and in particular their organizational impact, is also of interest to this thesis. Gantt and Nardi [Gantt92] concluded from their studies that there is generally a pattern of dependency upon a local expert among CAD users, a pattern CAD designers need to take into account. Robertson and Allen [Robertson92] identified that the difficulties in gaining proficiency in CAD technology makes work done on CAD systems difficult to manage and is the main reason why CAD tools frequently have had less impact than anticipated in corporations. Sinclair [Sinclair87] stated that "if there is one criticism that can be levelled at current research into human-computer interaction in CAD, it is that most of it seems to assume that design occurs in a vacuum;" a more holistic view of the design process was therefore proposed.

The advantages of a more usable user interface in CAD systems was one of the original motivations for this thesis. Presently, CAD system training is achieved through manual exercises and seminars which must educate the user about both the domain and the use of the CAD system. A more usable interface will eliminate this long and expensive training period by allowing the user to easily perform specific tasks and, thereby, efficiently build up their domain expertise and CAD system proficiency.

3.1.9 HCI Literature on CAD

The HCI community has long seen the challenges that CAD systems present and has made efforts to address them. However, as CAD continues to grow, more issues present themselves and it has become more and more difficult for the HCI community to keep pace without dedicating a specialized area of research into the HCI of CAD.

Majchrzak et al. [Majchrzak87] discussed the perceptual, which deals with geometric representation, and the cognitive, which deals with decision making, aspects of CAD. Sighting a study by Card et al. [Card80], Majchrzak et al. presented empirical data which identified CAD commands as not being a single operation but a combination of several commands. Ullman et al. [Ullman90] studied the importance of drawing in the mechanical design process. The results of the study pointed out the need for sketching in a CAD system, also identified by Tsang [Tsang86], and proposed that future CAD designs need to be driven from the "D," as in design, and not the "C," as in computers, aspect of CAD. Carroll and Stanley [Carroll86] presented a case study on the designing of a user interface for a circuit board design application. The study focused on visual texture perception and the presentation of features in the interface design. Carroll and Stanley concluded that graphic presentation plays a significant role in the total design of software systems and more research is needed to develop well defined principles.

John [John87] and the HUSAT team developed a model of the mechanical engineering design process which, pictured as a cosine, oscillates between administration and creation. The model offers a framework for evaluating CAD systems in terms of task coverage and an opportunity to define user oriented CAD systems as opposed to technology driven. The model also emphasizes the need to build CAD systems that support unobservable human behavior as well as that which is observable ("providing for Computer Aided Design in contrast to Computer Aggravated Draughting"). Jones et al. [Jones91] described the model based approach of problem solving as a "universal method ordinarily used by human beings" and computers which allow humans to focus their attention on the problem without worrying about describing the problem solving method to the computer.

van Zuylen [van Zuylen91] listed guidelines to support the end user and the user interface requirements for the interface of scientific programs. The emphasis of the work was on understanding the user and the problem domain of the application. Warchat [Warschat91] developed the CAD system MOCAD which extended a solid modeler by the following features: product design with a knowledge-based user-interface, internal storage of non-geometric product information in an object-oriented data structure using Prolog, automatic determination of subassemblies and assembly sequences through knowledge-based modelling functions using Prolog, and automatic association of subassemblies to assembly families and thereby to existing assembly cells. This extended CAD system aids the designer in optimizing the product from an assembly standpoint and at an early design phase.

Rogers [Rogers88] conducted an experiment on three types of explanations for an expert CAD system, namely, rule-based, condition-based, and hybrid rule- and condition-based explanations. The results showed that the level of user satisfaction was dependent on the type of explanation provided and that the hybrid rule- and condition-based explanations were the most satisfying and useful.

3.1.10 CAD Literature on HCI

Generally focused on the science rather than the interface, the CAD user interface has been more functionality than usability oriented. Although most CAD interface designers would comment that they follow HCI guidelines in their designs, there have been no studies in the CAD community to show the effects of incorporating HCI guidelines into the CAD interface design or a process to improve the usability of CAD interface design.

Zecher [Zecher94] stated that the user interface is of primary importance in CAD/CAM software. He commented on a few desirable characteristics of an interface and described the interface of an existing 2D CAD program named 2DCAD. Vershel [Vershel85] focused on four topics of how an user interface can be designer for a complex integrated engineering system: advantages and disadvantages of terminal independence, using color intelligently, making the system usable by both experts and novices, and making the user interface flexible. The need for research and scenarios were discussed for each of these topics.

Gossard and Serrano [Gossard85] developed an interactive preliminary design package: MATHPAK. This package allows the user to define physical variables of models and to interactively construct sets of equations relating those variables by menu selections. The program then computes the geometry to satisfy the desired dimensional values. This package provides a path from the preliminary design stage to that of design refinement and optimization. Thatch [Thatch87, Thatch88] also developed an interactive graphical preprocessor, MECHIN, to facilitate the design process. This work concentrated on the development of an easy to use and effective mechanism data entry systems for spatial mechanism analysis and synthesis.

Ellis [Ellis83] showed how the use of interactive methods can aid in the creation of 2D drawings used in conceptual design. A 2D geometry database and a set of display interface tools were developed. Sakurai [Sakurai82] described CAD techniques that allow the input of 3D solid models through their orthographic views, focusing on interfacing wireframe with solid modelling and providing designers with more natural ways of constructing solid models. Tsang's [Tsang86] user interface generates three-dimensional data from the user's two dimensional sketching of a conceptual design. He also pointed out the need for sketching in a CAD system.

The Albert Consulting Group conducted a study on the "usability" of I-DEAS Master Series [Albert93]. However, the study lacks merit because it really did not deal with usability issues such as ease of learning and high speed of user task performance, but simply recounted the CAD system and presented numbers on number of steps to complete a task and number of functions available in x numbers of picks. The study did not run subjects through benchmark tasks which would simulate actual use of the package. The study also lacks merit in that its benchmark evaluation was based upon a single operator from the consulting group.

3.2 Metal-Ceramic Brazing

Metal-ceramic brazing was chosen as the domain of the thesis application because of its non-trivial task of design parameters refinement and the need to focus upon both the design parameters and the geometric model. The complex brazing rules and the set of twelve design parameters that are necessary to define a braze joint [Bass95], makes metal-ceramic brazing a very specialized field that requires domain expertise on the part of the braze designer. Since few CAD applications are commonplace in conventional engineering practices, the subjects chosen for this study had little or no knowledge about metal-ceramic brazing. This simulated quite accurately how most users approach a CAD system, being novices in both the application domain and the CAD system.

Pattee [Pattee68] presented a report which reviewed the technology of joining ceramics and graphite to metals and other material. The report emphasized ceramic to metal joining as practiced in the electronic industry where most of the development originated. Pattee discussed both bonding theories and methods of evaluating joints as they pertain to material selection, joint configuration, surface preparation, and other facets of joining problems. Helgesson [Helgesson68] also presented a review of technical methods for metal-ceramic brazing, and in addition, conducted an experiment to investigate the bonding mechanism in metal-ceramic seals, and especially ceramics with very little or no glass phase. Akselsen [Akselsen92] reviewed the main parameters in the direct brazing process with primary emphasis on those influencing wetting of solid ceramics by liquid filler metals and concluded that wetting seems to be the limiting factor to obtain sufficient adherence in the joint. [Schwartz90] and [American63] comprehensively present the metal-ceramic brazing process.

Suga and Miyazawa [Suga89] presented a surface activation method which causes direct bonding between the metal and ceramic surfaces at room temperature. The sputter-cleaned surfaces are bombarded by ions which activates the surfaces which are then pressed to each other in a clean atmosphere. With this method, no thermal activation is necessary to obtain direct bonding between the metal and the ceramic if the absorbed species or inactive layers on the surfaces have been removed.

Bass [Bass95] is currently working on the design automation of the metal-ceramic brazing process. Specifically, fuzzy logic is applied to a rule base to recommend a set of optimal design and braze parameters. Bass's ongoing work formed the basis for the development of the metal-ceramic user interface.

3.3 Finite Element Analysis

There is an abundance of literature which presents comprehensive definitions and approaches to FEA ([Fenner75], [Davies80], [Zienkiewicz71], [Tong77], [Bowes75]). Most literature on FEA include basic principles (virtual displacements and element stiffness matrices) and element considerations (plate, shell, and solid elements).

Sabonnadiere and Coulomb [Sabonnadiere87], another general text on FEA, furthermore described the general architecture of CAD systems that support the finite element method.

Finite element analysis is one of the most common CAD applications because of the limitations of conventional stress techniques in analyzing irregular geometries. Even so, no usability studies have been found on this application. The usability evaluation of I-DEAS' FEA module in this thesis provided a comparison for the metal-ceramic interface with respect to interaction styles, presentation of domain knowledge, and task performance as well as a control comparison with the state-of-the-art in CAD user interface.

3.4 Observations

No studies which evaluated the usability of CAD systems or, more importantly, proposed solutions to usability problems with existing CAD systems, were found. Most books on CAD recognized the importance of human factors in the design of the graphical user interface. However, the discussion often digressed to listings of basic HCI design guidelines or description of existing CAD interfaces. Thus, it was stressed that it's sufficient to be sensitive to usability design issues, rather than that the achievement of usable interface is the result of usability oriented design process. In order to stress the latter, one of the purposes of this thesis is to introduce formative evaluation to CAD system development so that CAD user interface can be systematically engineered for usability.

CHAPTER 4. Designing and Testing the Interface

Although metal-ceramic brazing has been studied since the 1930's, the computer automation of this process has still to be developed. The metal-ceramic brazing user interface developed for this thesis was based upon the twelve design parameters identified by Bass [Bass95] and the intention of a future integration with a general CAD system. The effective manipulation of the twelve design parameters is the primary concern of the interface because they specify the metal-ceramic braze: (1) metal material, (2) ceramic material, (3) filler material, (4) cost, (5) operating temperature, (6) braze strength, (7) braze stress, (8) braze technique, (9) braze atmosphere, (10) braze heat schedule, (11) gap size, and (12) the geometry of the braze surfaces. Other attributes such as coefficient of thermal expansion (CTE) and ultimate strength of the materials are also of concern.

4.1 Interface Purpose

The metal-ceramic brazing interface is intended to be one module of a future CAD system that completely supports the design process. In this CAD system, the designer would access engineering applications, existing projects, customized and standard parts, and design rules, as needed. Since the completion of a typical design project usually requires many engineering processes and parts, the intention was to create a CAD system module that would allow the user to complete a project entirely within this one CAD software package. In essence, it would encourage a more holistic approach in which the user can transfer a design from one application to another.

The metal-ceramic brazing interface is also intended to be a practical tool, containing an expert system, that allows a user to design a basic metal-ceramic braze with the help of the system. Because metal-ceramic brazing is a very specialized field of study, a user is expected to have minimal brazing knowledge and some basic engineering knowledge. The purpose of the interface is not to educate the user about metal-ceramic brazing, but rather to provide a practical tool to help the user to design a workable braze with minimal user input.

The purpose of the expert system is to ensure that a good braze is still achieved even if the user has little or no knowledge in metal-ceramic brazing but has a specific brazing task in mind. The system should also accommodate those that are learned in the metal-ceramic brazing process and would prefer to explicitly specify all aspects of the braze design themselves.

4.2 The Initial Metal-Ceramic Brazing User Interface Design

The initial interface was designed such that the parameters were displayed at all times and allowed the user to manipulate them with simple selections (Fig. 1). Each design parameter, except for the geometry of the braze surfaces, was represented in an individual menu frame with the name of the parameter at the top of the menu. A field in each menu displayed the current selection for each parameter. The parameters which were designated by numbers (operating temperature, braze strength, braze stress, cost of braze, braze temp, and the CTE and ultimate strength of both metal and ceramic), had up and down arrows next to the display field which allowed the user to manipulate the parameter's magnitude. Those parameters which were designated by a given name (metal, ceramic filler, braze technique and braze atmosphere), had a "..." button next to their display (Fig. 2). When pressed, the "..." button selection would display a selection dialogue box which contained the selections (Fig. 3).

The selections in the selection dialogue box were mutually exclusive, only one could be selected at once. Those selections which were not compatible with the constrained parameters were grayed out, however, they were still selectable. If selected, another dialogue would pop up on top of the selection dialogue box to explain why this selection was not available. The "Not Available" dialogue box also gave the user two options: "Dismiss" and "Override." The user could choose to accept the fact that this selection was not available and design the braze around this selection or override the rule which was excluding this selection. The selection dialogue box allowed the user to dismiss the dialogue without making a selection or after a selection was made. After a material selection was made (i.e., metal, ceramic, filler), the "Vendors" button would become available and, if selected, a dialogue box listing all the vendors carrying this selection would be displayed (Fig. 4).

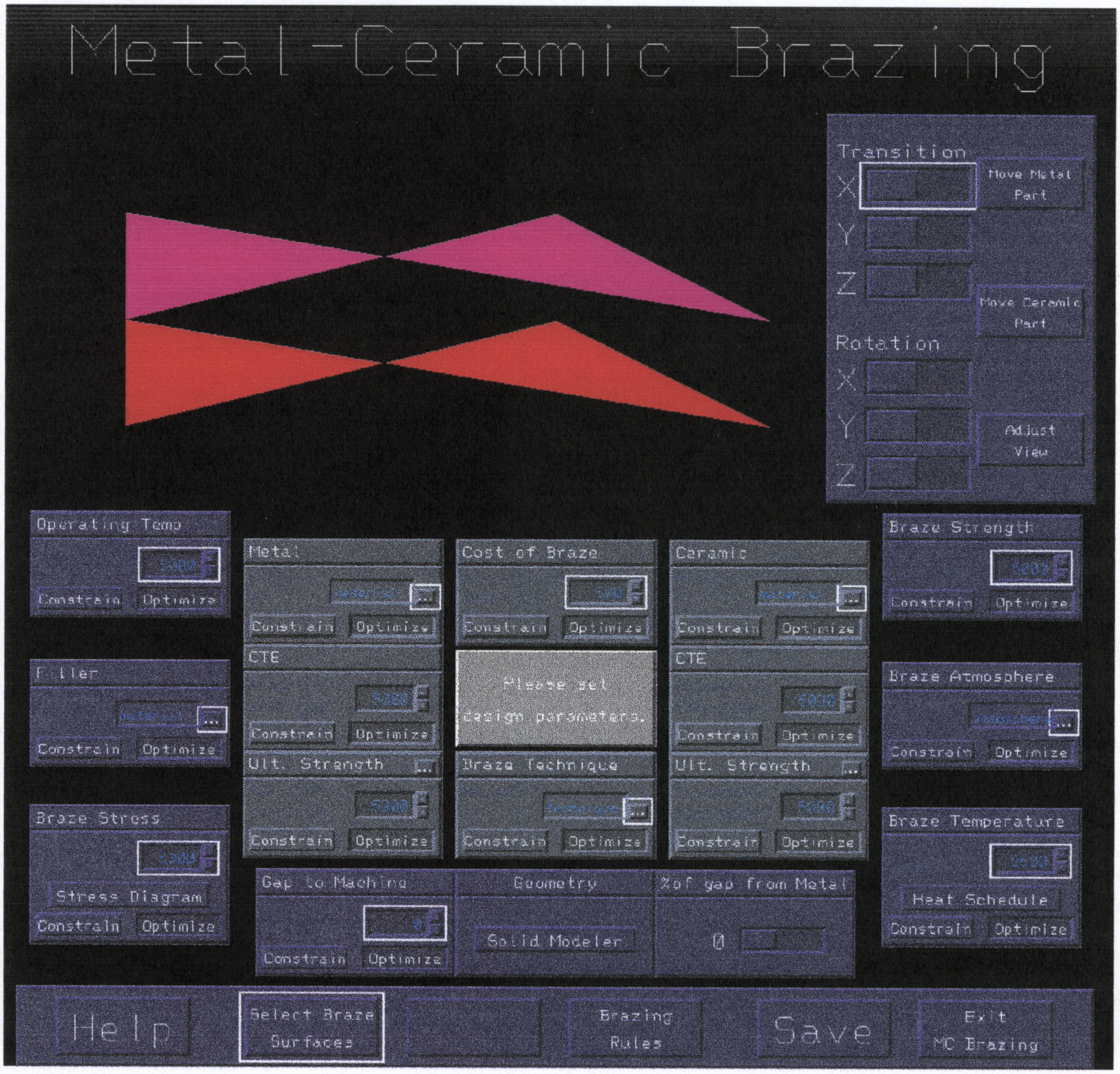

Fig. 1 The initial metal-ceramic brazing user interface.

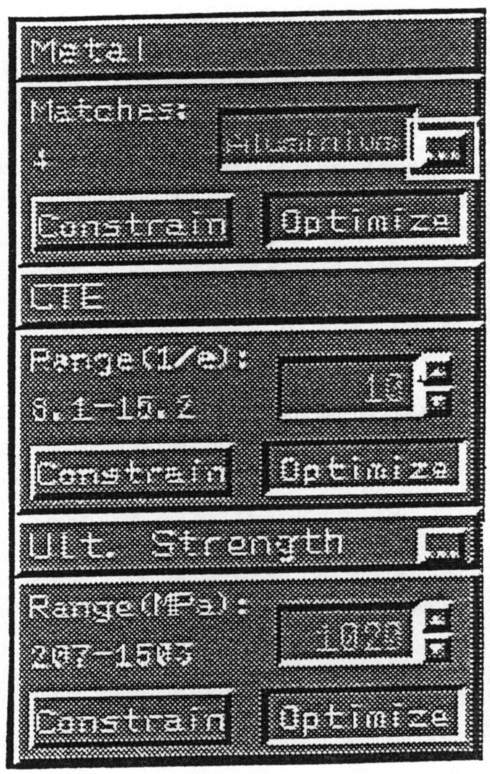

Fig. 2 The "Metal" menu frame.

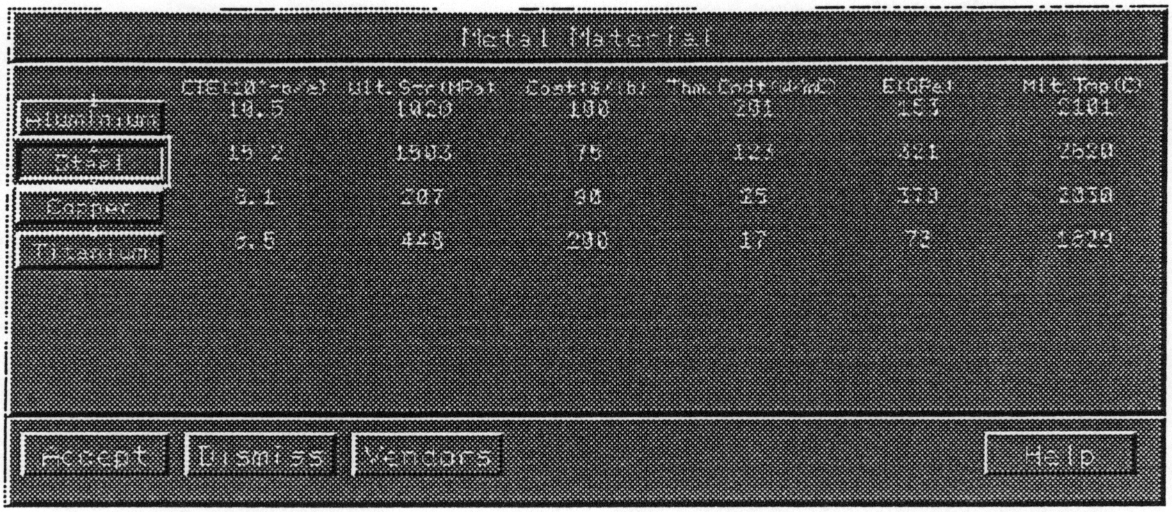

Fig. 3 The "Metal Material" dialogue box.

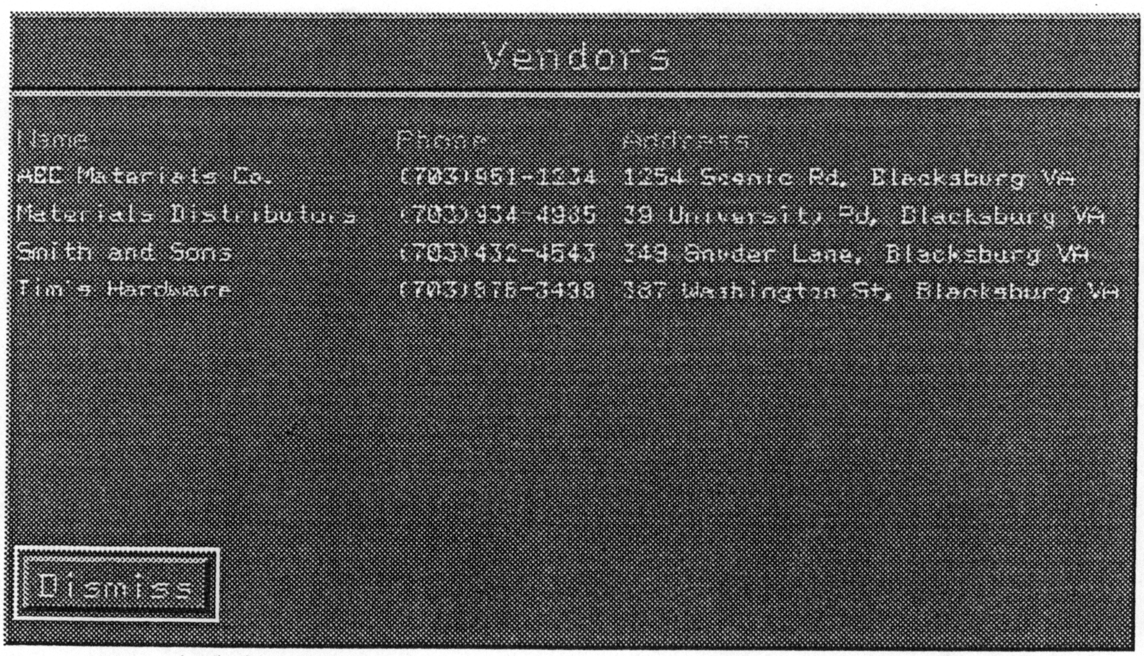

Fig. 4 The "Vendors" dialogue box.

The selection dialogue box also contained an "Accept", a "Dismiss", and a "Help" button. The "Help" and "Dismiss" buttons are available at all times, while the "Accept" button is only available after a selection is made. The "Dismiss" button would undisplay the dialogue box, but would do nothing. The "Accept" button, however, would undisplay the selection dialogue box and set a specified parameter as displayed in the display field.

An "Optimize" and a "Constrain" button were located at the bottom of each menu frame. The "Constrain" button constrained the parameter to a user-specified selection as specified in the display field. The "Optimize" button, when selected, had the system select the best selection for this parameter, which then was displayed in the display field. The optimization was based upon the constrained parameters and the consequent selections listed in the selection dialogue box. These two buttons were mutually exclusive since they were intended as status selectors. The "Constrain" button would change its color to red to indicate the constrained status, and back to its original color when unselected. The "Optimize" button, however, retained its color regardless of whether it is selected or not. Red was used for the constrained status to highlight its prominence. Because the constrained parameters are those selected by the user, they should stand-out from the other parameters such that the user can distinguish them from other possible parameters.

Another feature of each menu frame was the match or range. This feature indicates the number of matches or the permissible range of the parameter given the current constraints. The match or range were only visible when the parameter was optimized. This feature allowed the user to have a good idea of the limits of the available selections when deciding upon an unconstrained parameter.

Because of the fact that all parameters affect all other parameters through these hub parameters (metal, ceramic, cost of braze, and braze technique), it was considered to be important to communicate this relationship to the user. To promote the hub and wheel relationship between the parameters, an icon was displayed in the center of the parameters (see Fig. 1). The icon was a miniaturized version of the parameter layout with lines stemming from the hub parameters to the other parameters. Thus, the icon not only served as a possible icon to access the metal-ceramic brazing application in the main system for future versions, it also served to educate the user about how the parameters related.

4.3 Formative Evaluation of the Initial Design

A key focus of this thesis is the application of formative evaluation in CAD software. Because of the lack of previous studies, there existed no paradigm for an effective evaluation of CAD systems after which this work could have patterned itself. Therefore, a scenario which places the subjects in a situation that would most closely simulate the usage of the software in actual practice was devised. To address the issue of intuitive learnability, for novice users in unfamiliar CAD systems, the scenario casted the subjects as new employees at an engineering consulting firm. They were asked to perform tasks on a client's project, with which they were unfamiliar, on the firm's engineering software (the metal-ceramic brazing user interface and the I-DEAS finite element interface), with which they were also unfamiliar. The subjects were brief on the engineering theories behind the benchmark tasks, but no additional information was given on how to operate the software. The subjects were asked to relied only on their intuitive interpretations of the user interface in using the software to answer questions pertaining to the designs and some basic questions which were key to the software's purposes.

4.3.1 Developing the Experiment for the Initial Design

Qualitative data was collected for the purpose of pointing out and remedying design flaws which led to the users' difficulties in understand the purposes and the domains of the applications. The intent was to study how well each package supports basic tasks that a novice user may be asked to perform on such a system instead of evaluating the usability of the entire metal-ceramic brazing user interface or the I-DEAS finite element interface. Hence, the evaluation sought to understand the users' comprehension of the domain of the interface and the flow of the design process as presented by the interface, and to understand the user's command of the interface to perform the basic tasks and to answer important design questions.

4.3.2 Benchmark Tasks for the Initial Metal-Ceramic Brazing User Interface and the I-DEAS Finite-Element Module

The benchmark tasks for the metal-ceramic brazing user interface asked the subjects to work on a metal-ceramic joint of a ceramic engine that is under development. The subjects were given two sets of tasks to complete. The first set of tasks was a list of very specific tasks that took the subjects through the basic concepts such as constraining and optimizing the design parameters and the expert support of the system (Fig. 5). The second set of tasks presented a change in the braze design and asked the subjects to answer a few questions based upon that design change using the metal-ceramic brazing user interface (Fig. 6). Each set of tasks contained an introductory paragraph which briefed the subjects on various design situations. After the subjects completed the tasks, a questionnaire was administered to collect further information about the user's impression and understanding of the interface (Fig. 7).

The benchmark tasks for the I-DEAS finite-element module was patterned after the benchmark tasks for the metal-ceramic brazing user interface. Two sets of tasks were given to the subjects where the first set was a list tasks with very specific instructions and the second set presented questions pertaining to a design change (Fig. 8 and Fig. 9). Similarly, a questionnaire was also given to the subjects to ask about their impressions and understanding of the interface (Fig. 10).

You are a new employee at an engineering consulting firm. You boss has asked you to work on a metal-ceramic joint of a ceramic engine that one of your company's client is developing. The names of the two parts that you are to braze together are "Metal Part" and "Ceramic Part." Your boss wants you to perform the braze following the steps below and to answer the accompanying questions about the joint.

1. Start the "MC Brazing" application with the customized parts, "Metal Part" and "Ceramic Part."

2. Select the part on the top as the metal brazing surface.

3. Select the part on the bottom as the ceramic brazing surface.

4. Constrain the metal part's material to steel.
   How many ceramic materials can be used with Steel?
   What are they?
   How much do they cost and from whom can we buy them?

5. Optimize the metal part's material and constrain the ceramic part's material to Molybdenum.
   How many metal materials can be used with Molybdenum?
   What are they?
   How much do they cost and from whom can we buy them?

6. Which fillers can be used with Steel and Molybdenum?
   According to the computer, which is the best filler for Steel and Molybdenum?

7. Constrain the filler to the best filler for Steel and Molybdenum.
   How much does the braze with Steel, Molybdenum, and this filler cost?
   How strong is the braze with this filler?

8. Which "Braze Methods" and "Braze Atmospheres" can be used with Steel, Molybdenum, and this filler?
   Which Brazing Methods and Atmospheres are available to us?

9. Optimize the filler and constrain the cost of the braze to $200.
   Describe the best braze we can make with Steel and Molybdenum at $200.

10. Save this braze as a Project under the name "Best Braze"

11. Exit MC Brazing.

Fig. 5 The first set of benchmark tasks for the initial metal-ceramic brazing user interface.

Your company's client has decided to change the ceramic part's material from Molybdenum to Zirconium. Your boss has asked you to answer the following questions about the joint with this new material:

1. How much does Zirconium cost and from whom can we buy them?
2. Is Zirconium compatible with Steel? If not, select Steel to find out why not?
3. Describe the best braze we can make with Steel and Zirconium.
4. Describe the best braze we can make with these two surfaces.
5. Save this braze as a Project under the name "Best Braze"
6. Exit MC Brazing.

Fig. 6 The second set of benchmark tasks for the initial metal-ceramic brazing user interface.

1. What are your general impressions of the package?
2. What did you like or dislike about the package, please be specific?
3. Were you able to understand that the parameters were related and how they are related?
4. Did you think there was too much information on the screen for you to digest?
5. Did you understand the parameters for the metal-ceramic brazing processes?
6. Anything else you would like to add?

Fig. 7 Questionnaire for the initial metal-ceramic brazing user interface.

"testing.mf1" is a critical part in the ceramic engine's design. Your boss has asked you to perform a finite element analysis on the part following the steps below and answer the accompanying questions (Please *do not* save any of your work during the session). Before you start, here's a few hints to help you along the way: The "F1" button with mouse movement will move whatever is displayed. The "F2" button with mouse movement will zoom in and out. The "F3" button with mouse movement will rotate whatever is displayed. Hold down "Shift" key to select multiple nodes or surfaces. Click the middle mouse button or the "Enter" key for "(Done)".

1. Start I-DEAS with the "testing.mf1" file, "Simulation" application and "Meshing" task.

2. Create a FE Model with the geometry on the screen and "Fem1" as the model name. (Click on the mouse icon button and draw box around the part, by holding down the left mouse button, to select part. Name the part "Part1.")

3. Generate a mesh for the part and display the nodes as asterisks.

4. Select the "Boundary Condition" task.

5. Create a clamp restraint for all the nodes on the bottom surface. (Nodes must be selected individually. Move the part around to help you get a better view.)

6. Create a force of 15e9 in the y-direction at the center of the front edge of the top surface.

7. Go to the "Model Solution" task.

8. Create "Solution Set 1"

9. Solve the model.

10. Go to the "Post Processing" task.

11. Select "Maximum Principal" as the "Component" for "Stress" and "Magnitude" as the "Component" for "Displacement" of the displayed results.

12. Display the results. (Hold down the right mouse button and select "All" elements)

    What is the maximum displacement?

Fig. 8 The first set of benchmark tasks for the I-DEAS finite-element module.

Due to the change in material, there is also a change in the force applied to the "testing.mfl" part. Please perform the finite element analysis again but with the point force equal to 10e9 in the y-direction.

Fig. 9 The second set of benchmark tasks for the I-DEAS finite-element module.

1. What are your general impressions of the package?
2. What did you like or dislike about the package, please be specific?
3. Were you able to understand all the procedures that were required to perform the analysis?
4. Did you think there was too much information on the screen for you to digest?
5. Anything else you would like to add?

Fig. 10 Questionnaire for the I-DEAS finite-element module.

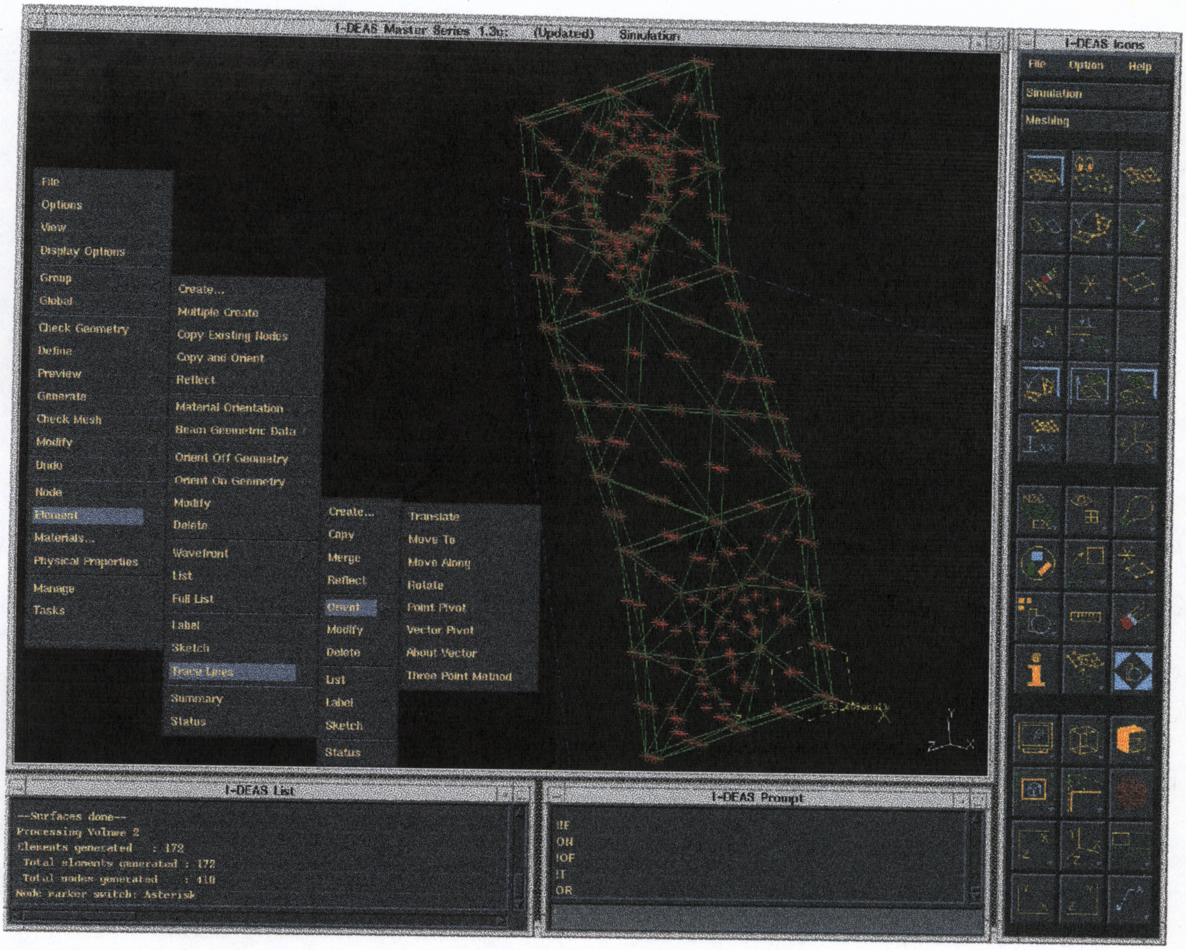

Fig. 11 The I-DEAS finite-element module.

4.3.3 Administering the Initial Benchmark Tasks

The test sessions were conducted in the Virginia Tech CAD Laboratory. The sessions were conducted according to the general procedure outlined in Section 2.2.6.2. The experimenter was at the side of the subjects throughout all of the sessions, taking notes on the subjects' interactions with the interface. One subject participated at each session utilizing a verbal protocol. The subjects were asked to talk through the tasks as if they were doing a "play by play" of their thoughts and actions. Initially, the session were also recorded by audio tape but, this was later abandoned for all sessions at the requests of some subjects.

During the first round of test sessions, the subjects were given both the metal-ceramic brazing user interface benchmark tasks as well as the I-DEAS finite-element module benchmark tasks. The two sets of benchmark tasks were administered in altering order with no apparent effect on the results. The sessions generally lasted from one and an half to two and an half hours, depending upon the speed and tenacity of the subjects. Because of the apparent display of frustration by the subjects, the experimenter terminated some of the sessions before those subjects completed all of the tasks. But, in general, the subjects were encouraged to complete the tasks and, when struggling, were given hints by the experimenter to help continue in the tasks.

A set of questions were also administered to the participants after they had completed the benchmark tasks for each interface. This questionnaire allowed the participants to reflect upon their evaluation and further comment upon the interface.

4.4 Redesign of the Metal-Ceramic Brazing User Interface and the Formative Evaluation of the Redesigned Interface

The usability problems collected from the initial formative evaluation were divided into three categories: high, moderate, and low importance (Tables 1, 2, and 3). The problems were placed into each category based upon their importance in conveying basic domain knowledge and design process, and in helping the users to complete the tasks with minimal domain knowledge. Each suggested solution was either implemented or not implemented depending upon their importance and cost of implementation.

The metal-ceramic brazing interface was evaluated again after the redesigns were implemented (Fig. 12 and Fig. 13). The benchmark tasks given to the subjects for the redesigned interface were the same as those for the initial design. The purpose of administering the same benchmark tasks was to evaluate both the effectiveness of the solutions implemented and to ensure that the changes did not incur additional usability problems. Furthermore, the benchmark tasks for the initial design proved to be an effective tool in teasing out the usability problems of the interface design and an accurate simulation of a real world situation.

The I-DEAS finite-element module, however, was not evaluated again because it was not redesigned.

Table 1 Usability Problems -- High Importance

Suggested Change	Implementation Cost	Actually Implement?
make title of menus larger in both the Metal-Ceramic Brazing screen and the main CAD screen	low	yes
state both the rules and the parameters with which a selection is not compatible in the not compatible dialogue box	low	yes
display "Please select braze surfaces" in the center verbal cue box when the application is first started	low	yes
make "Optimize" button green when selected	moderately low	yes
"Metal constrained to Steel" and similar text in verbal cue box to reflect user selection	moderate	yes
change the color of "Metal" menu to gray	moderately low	yes
change color of "Ceramic" menu to brown	moderately low	yes
change the color of the metal geometric model to gray	high--color for the geometric model was difficult to control	no
change the color of the ceramic geometric model to brown	high--color for the geometric model was difficult to control	no
add "Metal" next to the metal geometric model after the user has selected the model to be metal	moderate	yes
add "Ceramic" next to the ceramic geometric model after the user has selected the model to be ceramic	moderate	yes
turn-off highlights	low	yes

Table 1 Usability Problems -- High Importance (continued)

Suggested Change	Implementation Cost	Actually Implement?
move the material rankings to the left of the material buttons	moderate	yes
use black text on selectable gray-out buttons	moderate	yes
double click to start application	very high--toolkit did not have list field	no
change the color of the bottom menu bar to red	moderately low	yes
display the "Select Brazing Surface" dialogue box when the user selects the model without selecting the "Select braze surface" button	moderately low	yes
move the "Filler" menu where the "Cost of Braze" menu is	moderately low	yes
move the "Cost of Braze" menu where the "Braze Technique" menu is	moderately low	yes
move the "Braze Technique" menu where the "Braze Strength" menu is	moderately low	yes
move the "Braze Strength" menu where the "Filler" menu is	moderately low	yes
a filter which displays projects with 2 parts	high	no
make all text larger	high	no
redesign of main CAD screen to improve projects, parts, and application management	very high	no
gray-out other material selections according to other constraints even when this parameter is constrained	moderately high	yes

Table 2 Usability Problems -- Moderate Importance

Suggested Change	Implementation Cost	Actually Implement?
use "Braze Method" instead of "Braze Technique"	low	yes
use "Stress in Joint" instead of "Braze Stress"	high--menu was not large enough to fit title	no
use "Strength of Braze instead of "Braze Strength"	high--menu was not large enough to fit title	no
use "Please wait, your project is being saved." in the "Please Wait" dialogue box to be more specific	low	yes
make the "Vendors" button always active and ask the user to select a material with a dialogue box which contains all the available materials when user selects the "Vendors" button with no material selected	moderately high	yes
change icon in the center to better illustrate the relationship between the design parameters	high	no--remove icon from interface
more apparent indication of the brazing rules that are in effect	high	no

Table 3 Usability Problems -- Low Importance

Suggested Change	Implementation Cost	Actually Implement?
if the braze is already saved, do not ask the user to save again when exiting	high	no
make input text in "Save" and "New Name" dialogue box selected	high	no
allow the user to type in numbers for the number boxes	high	no

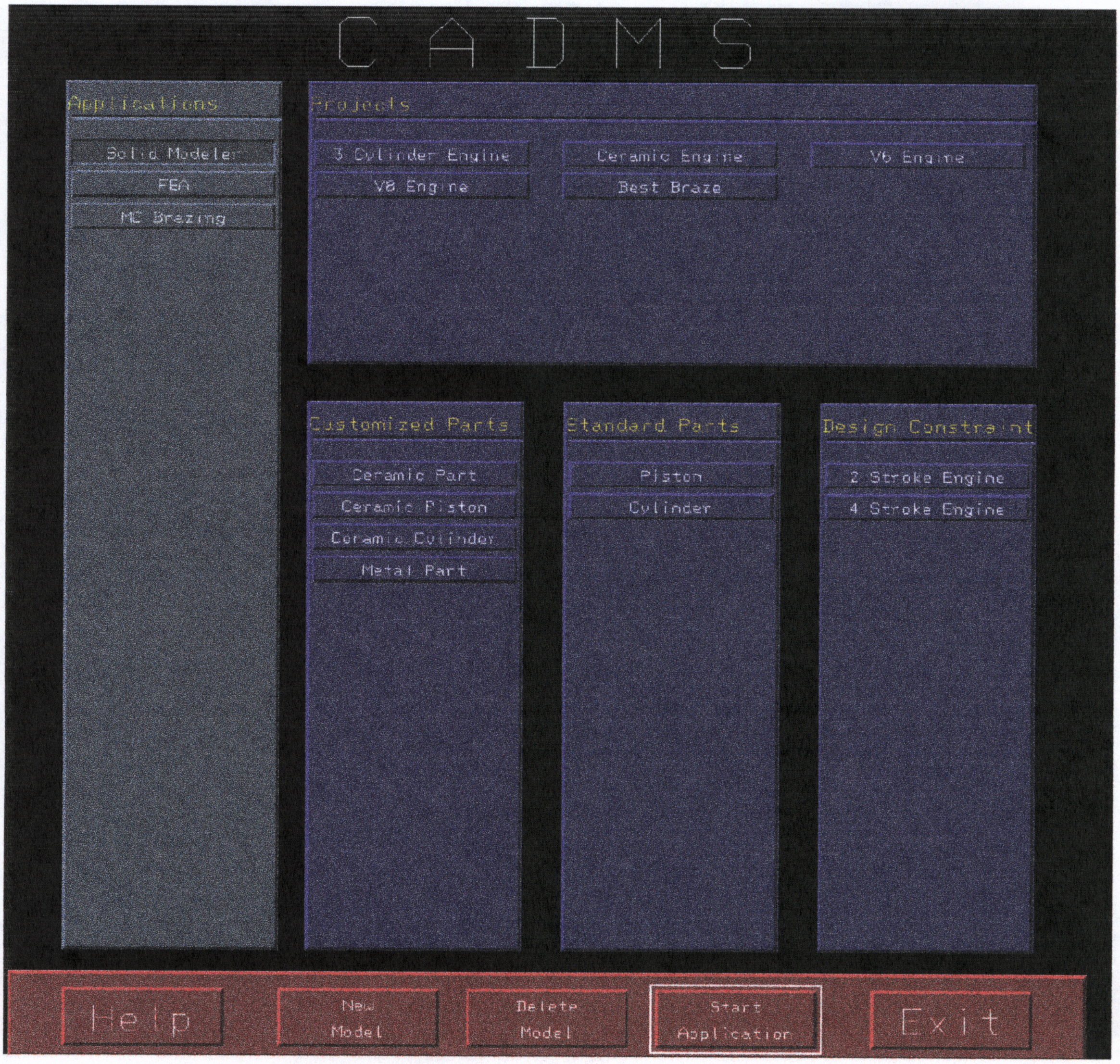

Fig. 12 The main CAD screen of the redesigned metal-ceramic brazing user interface.

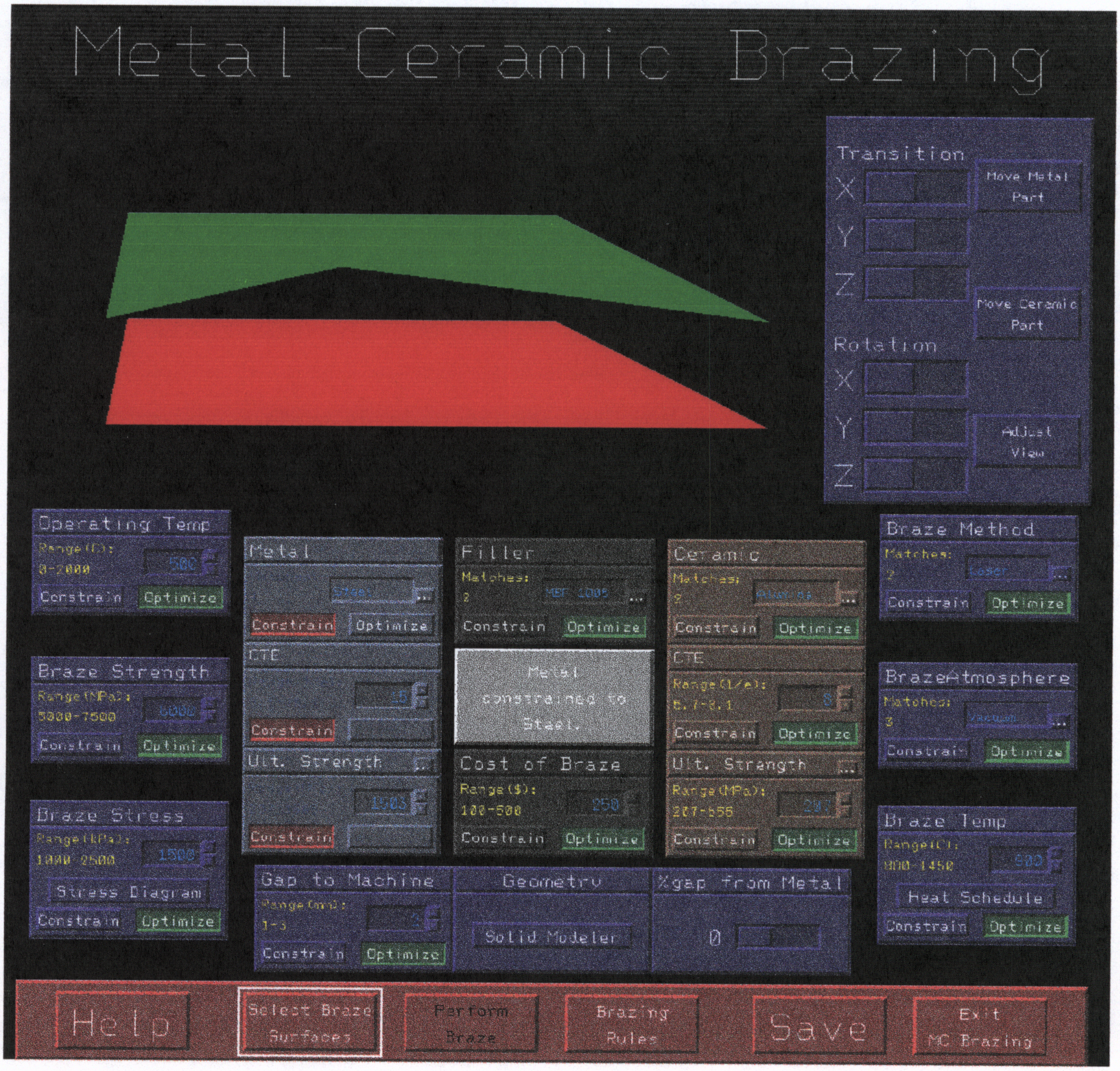

Fig. 13 The redesigned metal-ceramic brazing user interface.

CHAPTER 5. RESULTS

5.1 User Demographics

The target user group for this thesis was engineers who were novices in both the application and the application domain. The actual user group contained mostly mechanical engineers, ranging from college seniors through doctoral candidates, together with a few non-engineering subjects (Table 4). This was done to insure that there were users present with absolutely no application domain knowledge available to unbias the data collected from the engineering subjects. After testing two non-engineering subjects, no more non-engineering subjects participated in the study as their test sessions produced no usability problems different from those of the engineering subjects. The only difficulty that the non-engineering subjects had that the engineering subject did not have was with the engineering terms. When reading the "Braze Strength," the non-engineering subjects simply said "four hundred M-P-As" instead of "four hundred Mega-Pascals." They understood that "MPa" stood for a strength unit but did not know what unit this was.

All of the engineering subjects used computers on a daily basis and most had at least two months experience with a CAD package. None of the engineering subjects had any knowledge about metal-ceramic brazing while some had limited understanding of finite-element analysis. Only one subject had experience with expert systems. The test subjects' lack of experience with metal-ceramic brazing and finite-element analysis and familiarity with computers very accurately represented the target user group.

Although some subjects had previous finite-element analysis experience, this did not appear to be an advantage; only 2 out of 4 subjects completed the tasks involving the I-DEAS interface. In fact, this was by design. The benchmark tasks were designed to not require more than a basic finite-element analysis knowledge which was explained to each subject at the beginning of each test session. Hence, any advantage due to previous FEA experience was eliminated.

Table 4 User Demographics

Subject #	Academic Level	Major	Computer Experience	CAD Experience	Metal-Ceramic Brazing Knowledge	Finite-Element Knowledge
Pilot	PhD	ME	use daily	AutoCAD	not much	a little bit
1 (1st round)	MS	ME	use daily	I-DEAS, PCADAM, AutoCAD	a little	some
2	MS	ME	use daily	I-DEAS, PCADAM, AutoCAD	not much	fairly good
3	MA	ESL	use daily	none	nothing	nothing
4	PhD	ME	use daily	none	nothing	nothing
5	BS	ME	use daily	PCADAM, CadKey, AutoCAD	nothing	class project w/ simple software
6	BS	ME	use weekly	CadKey, PCADAM, Algor	not much	FEA class
7	BS	ME	use daily	PCADAM	nothing	some
8	BA	Education	use monthly	none	nothing	nothing
9 (2nd round)	BS	ME	use daily	CadKey, PCADAM	nothing	very elementary
10	MS	ME	use daily	I-DEAS, PCADAM	none	some
11	BS	ME	use daily	I-DEAS, PCADAM	none	some
12	PhD	ME	use daily	I-DEAS, Abacus	none	good
13	BS	ME	use daily	PCADAM, CadKey, AutoCAD	nothing	introductory level

5.2 Qualitative Data for I-DEAS Finite-Element Module

I-DEAS is a very well-developed and powerful CAD system. It utilizes menus and icons in a consistent layout to provide a highly interactive user interface. Many subjects recognized the power of the application. It is also clear that I-DEAS is a system that requires extensive training to use efficiently. The purpose of this study was to demonstrate that formative evaluation can be applied to a commercial CAD system and to explore the intuitive learnability of the system for a novice user unfamiliar with it, rather than to elaborate on its advantages and disadvantages. The reader should therefore not user the results of this evaluation as an indicator of the commercial or engineering value of I-DEAS. But rather, the results of this evaluation only indicate the difficulties first-time, untrained users had with the system.

Every test subject struggled with the I-DEAS interface. Six out of the nine test subjects, including the pilot, did not complete the benchmark tasks. "It's crazy to come in here and expect somebody to sit down and accomplish anything... Once I spend a couple hundred hours in front of the machine I would do great." "I can sit here for days just figuring things out." Perhaps the most telling of the users' problems with the I-DEAS interface are summed up in this quote from one test subject: "They have a reason for grouping things, but I just don't know what the reasons are."

5.2.1 Menus

"It's pretty difficult because of so many menus." "The menus are very frustrating." All the test subjects had difficulties using the menus and spent most of the session just searching through the menus, struggling to locate the desired menu items; in some cases, subjects never found what they wanted. This is a severe interaction problem because the I-DEAS interface is mostly menu driven and if the user cannot locate a specific menu item, the system becomes useless to the novice user.

The wording of the menu items was a major factor in the test subjects' problems with locating menu items. For example, "Manage" suggests the controlling or manipulating of something already in existence but "Create FE Model" is under "Manage." This was the cause of many problems during the test sessions. Only one test subject found "Create FE Model" under "Manage" and the other subjects gave-up and proceeded to the next task. Many subjects quickly lost confidence in the interface as they struggled to create a FE model, the first step of the I-DEAS finite-element analysis process. The only subject who found "Create FE Model" under "Manage" commented: "Problem is wording. What is 'Manage?'"

To specify restraints, the user has to select "Create" which is under "Restraint" which is under the "Boundary Condition" task. One user commented: "Where's restraint... I wouldn't think 'Create' as the entry path." This was further confused by the fact that there is a "Boundary Conditions" menu item under the "Boundary Condition" task. Many subjects selected the "Boundary Condition" menu item which called-up a dialogue box that displayed the boundary condition sets applied to the current model. The most frustrating aspect of this dialogue box was the fact that it did not allow the user to specify boundary conditions. The "Boundary Condition" menu item was extremely misleading as one subject stared at the "Boundary Condition" dialogue box wanting to set restraints:

"I'm lost. It says 'Restraint Set' but I don't know what I can do. I feel I'm in the right spot but what can I do?" Another subject struggled with the same problem and said: "I understand boundary conditions but if you click on 'Boundary Condition' you get a lot of stuff that sounds like they are buried in a manual."

There were also instances where the difference in menu items were ambiguous to the subjects. One subject commented: "I wonder what is the difference between 'Result' and 'Display Result?'"

In most cases, the wording of the menu items simply did not match with the tasks. This confused the subjects with respect to the grouping of the menus and left them to search through the menus for the items that best matched the task at hand. This is a serious problem because the test subjects did not understand most menu items, and hence, due to frustration, had given up trying to understand them. "Why would I find all this stuff under 'Task?'" one subject exclaimed. The lack of word association often left the subjects to second guess themselves and loose confidence in the interface. The following quotes of users looking for "Create FE Model" clearly illustrate this design flaw: "Generate? Where's create? Define? How about something with finite-element?" "How to create a model?" "I don't know how to create a model. I'm stuck. 'Element?'" "I can't find the key word [Manage]." "First thing that comes to mind is create, I want to make a model of it." "Okay, I give up, how do I start a model."

Another problem that contributed to the users' difficulties with the menus was that the menus were both too deep and too broad. The menus were six levels deep with ten to twenty menu items at each level. "I can't believe they went out that far!" one subject exclaimed after seeing all six layers of menus. Most subjects would complain "Something's wrong with the interface because you can't find what you want." This was a problem that's largely, but not exclusively, contributed to the menu system of I-DEAS. The long and deep menus tend to hide items from the users. In most cases, the subjects knew what they were looking for, but struggled to find the correct menu items that took them down the menu tree to that desired function. "No, I have no idea how to display the nodes as asterisks. This is when I would give up or call somebody. I honestly don't know what I'm suppose to do, I would just get out." "If what you are looking for is not there, you don't know how to find it." A usability guideline states that menu systems should be only three to four levels deep and four to eight items on a menu [Snowberry83]. The I-DEAS menu system demonstrates the justification for this guideline. With its depth, the I-DEAS menu system had a tendency to loose subjects as they were trying to locate a function within the six layered menu tree.

This problem is further compounded by the fact that the submenus were, in most cases, just as long as, if not longer than, the menu in the previous layer. Most subjects complained that the "Menus are way too long" and have "too much information." Subjects also found it extremely tedious to look through one long list after another, and they would often loose their focus and not look as carefully. It was not uncommon for subjects to not see what they were searching for even though the menu item was displayed. "I don't know why I didn't see that." and "Am I just not seeing it or what?" were commonly heard comments from the subjects as they searched through the menus.

Another factor that contributed to the subjects' difficulties in locating menu items is the combination of the long list of menu items along with the size of the text. One subject commented after a long unsuccessful search for "Asterisks" that the "text are small and it makes my eyes tired so I don't want to search them out."

A common critical incident during the test sessions was that users would try to dismiss the menus by clicking on an area outside the menus; a standard interaction style for most menu systems. But the menus in the I-DEAS interface could not be dismissed unless cancel or the blank space at bottom of the first menu was selected. Most subjects never figured out why they could not dismiss the menus. Although this difficulty did not necessarily interfere with the benchmark tasks, it was a point of aggravation that added to the overall frustration experienced by the subjects.

There were no simple solutions for the menu system dilemma in I-DEAS, having to balance the large amount of functions in I-DEAS with that of their organization. However, there were instances where a simple change of wording would alleviate a problem. For example: use "Set Boundary Condition" as the menu item that allows the user to set boundary conditions and "View Boundary Condition" as the menu item that allows the user to view the boundary condition sets applied to the current model. Simple changes like this would help to bandage some of the usability problems, however, the root of the problem remains unresolved. This issue is discussed further in Chapter 6.

5.2.2 Icons

In addition to the menus, I-DEAS offered icons as an alternate method of interaction. Besides serving as buttons to system functions, the icons in I-DEAS also interacted in a fashion that's very similar to menus. By holding down the left mouse button with the cursor over an icon button, a pop-up menu would appear with a list of more icons and textual explanation of each icon. The user could select an item from the list to access that specific function as in a menu. Upon dismissing the pop-up menu, the icon button would change into the icon that was just selected to reflect that a selection was made. Because this was not a standard interaction style for icons and there was no obvious visual cue to indicate that the icons also serve as a menu system, only one user figured out this additional feature of the icons.

Although all users agreed that the icons are a good interaction style, only two subjects used the icons, and then only because the menus had disappeared and the subjects did not know how to get the menus to reappear. Most subjects did not understand what the icons meant and just ignored them. "Don't understand the icons. [They are] hopelessly cryptic" "No, I have no clue what they [icons] are, these things are meaningless." "Gazillion icons, they look nifty, but I don't understand them." "These things I can make no sense of." "I like the icons, but I have no idea what they mean."

Most subjects suggested that the icons need some textual explanation and one subject suggested Macintosh-style bubble help as a solution to help the user understand the icons. As mentioned before, I-DEAS did offer textual explanation for the icons in the menu feature, but only one subject discovered this feature. However, according to this subject, the textual explanation were just as cryptic as the icons; in most cases, he did not understand the meaning of either the icon nor the textual explanation. "I thought I might know what the icons meant but I had to use the menu mostly."

The icons face a similar dilemma as the menus with respect to their large quantity and organization. This too is discussed further in Chapter 6.

5.2.3 Focus

To help the user to understand the interaction, I-DEAS offered the "I-DEAS Prompt" window, a text input window located at the bottom right portion of the screen, next to the "I-DEAS List" window, a text output window. Although both these windows served important purposes in the interface, they were relatively small in size in comparison with the menu and the icons. Not only were the windows small, the text that were displayed in them were also small. Users would often ignore them because these windows would just get filled with long lists of text when tasks were performed. Usually the subject just didn't understand what was displayed in them and therefore ignored the whole bottom of screen.

The "I-DEAS Prompt" window and the "I-DEAS List" window worked hand-in-glove with the menus and icons to help the user understand the interaction. They prompted the user to select elements and asked questions about the selection as well as reflected what I-DEAS had done in response to user selection. However, these windows had to compete with the menus and the icons for the user's attention as they were on different part of the screen. These smaller and more complex windows, which also required more reading, often were not noticed and the users were lost again. One subject commented on the I-DEAS Prompt window: "[I] liked the prompt but it was the last place I looked definitely.

The following illustrates this problem: I-DEAS dismissed all the menus when "Restraint" function was selected and prompted the subjects in the "I-DEAS Prompt" window to "Select element(s)." Most subjects only noticed that the menus were gone and did not see the query in the window. This caused confusion and panicked attempts to make the menus reappear. "I've lost everything. How do I get the menus back?" By enlarging the size of the text and only displaying information relevant to the task at hand, the system could encourage users to take more notice of these windows. It would also help if the "I-DEAS Prompt" window would highlight or show itself to be more prominent when it had a query for the user.

Another focus problem was the placement of the dialogue boxes. The dialogue boxes are usually complex and require much attention from the users. They appeared over the top of the icon menu bar which was usually more complicated than the dialogue box and had, in addition, more colors. The combined busyness of the dialogue boxes and the icons below often caused subjects to miss items in the dialogue boxes. A simple solution would have been to display the dialogue boxes away from the icon menu bar and more toward the center of the screen.

5.2.4 Inconsistency

The most glaring inconsistency encountered during the testing was that of the grouping of the menu items. Trying to generate a mesh for the part, the subjects would select "Generate" and then "Mesh." A third menu would appear after the selection of "Mesh" with "Part" listed along with element types such as "Thin Shell" and "Solid Mesh." The reasoning behind the selections are not consistent because they refer to different specifications of the mesh. The reason behind the menu item "Part" is to generate a mesh for the part. If "Part" was selected I-DEAS would automatically generate a mesh for the geometric model. However, the other selections refer to specific element types. If one of these was selected, I-DEAS would generate a mesh with this element type for the part. Most users never caught on to the meaning of "Part" and selected one of the element type which led to a dialogue box of queries and specifications.

To better indicate what each selection actually meant, the addition of "with" in front of the element types and "for" in front of "Part" could be helpful. This would result in the menu selections "Generate", "Mesh", and "for Part" to generate a mesh for the part and "Generate", "Mesh", and "with Thin Shell" to generate a mesh with thin shell element.

5.2.5 Visual Cues

I-DEAS requires the user to perform finite-element analysis in a predetermined number of steps and order: (1) create FE Model, (2) mesh the part, (3) set boundary conditions, (4) create solution set, (5) solve, and (6) display results. However, I-DEAS does not make it obvious to the user that one must proceed in this fashion. Hence, one of the most common complaints, and a major cause for why subjects did not finish the benchmark tasks, was that I-DEAS offered no sense of the procedure required. "What to do next?" "Frustrating to not know anything and have to do anything." "It's nuts to expect someone to come in and use this stuff." "What's the next step, let me think about." "What's next?" "There's no trail to follow." "I want to go on, but I don't know what to do next." "Without the instructions [the benchmark tasks], you would find yourself in the middle of the woods."

All the subjects who reached the solving stage of the finite-element analysis procedure commented that solving takes a long time. While lines and lines of text scrolled by in the I-DEAS List window, the subjects just sat there wondering how much longer it would take. I-DEAS did change the cursor to appear as a watch, hinting for the user to wait. This was usually a good feedback for a short delay, but for a wait that took a few minutes, the user was just left hanging. A better feedback device would be a time-remaining-in-process or percent-of-task-completed indicator. This would give the user a time frame of how long the system will take to complete the process, and it would avoid leaving the user hanging.

I-DEAS allowed the user to rotate the geometric model by holding down the "F3" function key while moving the mouse. Although there were no visual cues to indicate the "F3" key's function, it is a quick tip that the user can be informed of before starting to use the system. The test subjects were all informed of this feature and remembered to access it when it was needed in the benchmark task. However, there were no visual cues to indicate that the geometric model rotated differently depending upon where the cursor was when the "F3" key was pressed (the model would rotate in 3D if the cursor was close to the center of the graphics window and rotate in 2D if the cursor was close to the boundary of the graphics window). This caused quite a bit of confusion and some subjects were not able to rotate the geometric model to as desired. Some of this confusion could have been alleviated, had another button perhaps "F4", had been assigned to one of these modes, or had a more definitive message that indicated the two "F3" modes.

5.2.6 Feedback

Clear and understandable feedback is important in keeping the users' confidence in the interface. Timely and proper feedback also helps the user learn and understand the interface and the application domain. However, when the system doesn't react as users might expect, they usually start to second guess themselves and proceed with little or no logic. This was quite a frequent occurrence during the test sessions. For example, when selecting nodes for "Restraint", many of the test subjects would misinterpret the circle around the nodes as meaning the nodes were restrained. Most subjects expected highlighting to mean that the object have been selected. I-DEAS also used highlighting to show that lines or surfaces were selected. However, it used circles about the nodes when they were selected. This inconsistent treatment of the nodes confused the subjects and usually caused them to believe that they had accomplished the task of setting restraints, when in actuality, they had not.

I-DEAS changed the appearance of the cursor to be that of a watch to indicate that the user should wait for it to finish processing. Most users understood this but with some hesitation: "Is that a watch? Telling me to wait?" Enlarging the size of the watch so that it could be more easily recognized would alleviate some of this hesitation.

There is one point where I-DEAS behaves radically with no explanation to the user. During the long initial start-up, I-DEAS did not change its cursor into a watch. Instead, the I-DEAS main window would just disappear and reappear after some time. All the test subjects exclaimed "What happened?" as they saw the window disappear.

5.2.7 Handling Errors

Although I-DEAS allowed the user to back-up in geometric selections (e.g., nodes and elements), it did not allow the user to undo the last function (e.g., "Restraint" and "Create FE Model"). This caused some subjects to exit and restart I-DEAS or reopen the file whenever they believed they had made a mistake. "I want to quit and start over again." One subject commented that "most learning comes from making errors and going backwards;" I-DEAS, however, does not allow the user to easily do this.

I-DEAS relied greatly on error massages when errors occur. However, most test subjects the messages difficult to understand. "What is that [error message -- a part has no geometry] suppose to mean?" The error messages were also not carefully read because they usually just tell the user what the system could not do and did not give the user instruction as to how to correct the situation. The subjects would get into a habit of ignoring the error messages and sometimes missed those messages that did tell them how to correct the situation. A better method of interaction would be to gray-out those selections that the system could not perform instead of relying on error messages. For example, "Create" under "Solution Sets" should be grayed-out whenever boundary conditions have not been set.

5.2.8 Geometry Manipulation and Selection

In a graphic intensive program, such as I-DEAS, it is important that the user is able to easily manipulate and understand the graphical representations. This was made evident during the test session as the test subjects spent a good part of their time moving about the geometric model in order to pick nodes. Unfortunately, the subjects frequently had difficulties interpreting the geometric model since the 3D wireframe usually have so many lines. "I assume that selecting [geometric model] is a universal concept in this program that I just don't understand." Besides using shading as the method for users to correctly interpret the geometric model, not displaying lines that were behind the part or making them hidden lines would also help. This technique could also be applied to nodes.

It should be noted that the limitation of the hardware contributes to much of the graphical representation problem. Graphical representation methods such as shading requires addition computation time. This requirement can greatly reduce the speed of the software's response time and such techniques are often used sparingly to save users of waiting time. Balancing technical capabilities and user interaction needs presents an area for further usability studies.

Although I-DEAS allowed the user to select a part by drawing a box around the desired part, it did not allow the user to use the box to select nodes. Instead, the user had to select them one at a time while holding down the "Shift" key.

5.2.9 Anticipating the Most Likely Next Task

A system's usability depends greatly upon its ability to anticipate the most likely next step the user will take. Unnecessary steps to complete a task or, in more severe cases, losing the user's confidence can result from the software's failure to provide the next logical step. For example, the nodes did not display as asterisks when the subject selected "Asterisks" under "Display Options" and "Nodes". "I expected asterisks to show up after I selected asterisks, but it didn't, and now I don't know what to do again." It was logical that the user would select "Asterisks" only if they wanted to see the nodes displayed as asterisks. To require the user to make an additional selection showed that the system did not anticipant the flow in the users' reasoning and caused an interaction error. Many test subjects never completed this task as they did not think that they had to make another selection. "I don't know how to show nodes." "Show, show your darn nodes!"

After the solution of the part has been calculated, the subjects was required to first go to the "Post Processing" task, then select "Display Result", and finally select the part to view the results. This is a logical progression; after waiting for the system to process the solution, the user would want to see the results. Many steps could therefore be eliminated if the system automatically went to "Post Processing" and displayed the results if the solution was calculated without errors.

I-DEAS also failed to anticipate the next most likely step by asking the user to input a selection when there is only one selection available. The system should anticipate this selection and set it as the default. For example, when creating an FE model, the subjects were asked by I-DEAS to select the part for which they wanted to create an FE model. Since the displayed geometric model was the only part in the file, it was logical that the subject would select the displayed geometric model.

5.2.10 Help

It was not the focus of this thesis to study the on-line help facility of I-DEAS. However, the following quotes make a strong case for future studies in improving on-line help: "'Help' is not very helpful, manual would be more helpful than on-line help", and "'Help' hardly ever helps."

5.2.11 Miscellaneous

This section lists usability problems that did not fit under more a definitive usability category. They are presented here as instances of difficulties that the users had.

The default value displayed in the force magnitude entry field was "0.00000000." It was troublesome to have so many zeros after the decimal because the subjects had to erase all the zeros since the default value was not replaced automatically by the new value entered by the subject. This also caused some uncertainty as the benchmark task presented the force magnitude in scientific notation. The subject hesitated to enter the force magnitude in scientific notation since the default value displayed did not assure the subject that scientific notation would be accepted. "I'm not sure whether it will take this nomenclature."

Another point of confusion was that none of the results displayed have units. "I can tell you the magnitude but, what's the unit of this thing [results]?"

5.2.12 General Impressions

One of the post-benchmark task questions asked the test subjects of their general impressions of the I-DEAS interface. The following is a collection of subject response: "It was frustrating. But once I figured it out and started to see the engineering aspects of it, it was fun. I was like a boy playing with a sophisticated toy." "I found this one impenetrable." "I don't want to use this software anymore." "Golly this is so easy, not!" "It's hard to use." "When you said this wasn't intuitive, boy you weren't kidding." "It looks like its pretty powerful but it's really hard to use." "Hate to say it, but this is [kind of] frustrating." "Looks like a pretty program once you know how to use it!" "This is [going to] take a while, I don't know what I'm doing." "I'm learning [nothing] about this software." "Clueless on this. Oh boy!" "I don't even know how to get started." "I think it stinks." "It looks great. I just don't know how to use it. It's pretty frustrating not having someone or a manual telling you to do something."

5.3 Qualitative Data for Initial Metal-Ceramic Brazing Interface

The same eight subjects who participated in the I-DEAS FEA module evaluation were also invited to test the metal-ceramic brazing interface. All but one test subjects were able to correctly complete the metal-ceramic brazing benchmark tasks. The one test subject did get through the tasks but interpreted the interface feedback differently than as it was intended. The fact that this interface was a prototype that had yet to be refined brought about some difficulties during the test sessions. This was explained to all the subjects and they were very understanding and cooperated accordingly.

5.3.1 Layout

There were conflicting responses to the menu layout of the metal-ceramic brazing interface. Most of the subjects' initial responses were that the screen seemed cluttered. "Pretty confusing, so many buttons here." "This layout is fine but there's too much information on the screen at once." However, some of the subjects thought otherwise. "Not at all, not too much info on screen." "I think it's well organized." "Doesn't seem cluttered."

The consensus among the subjects seemed to be that there was a large amount of information on the screen and that it was difficult to search through all the menus. "Too cluttered, there's too much information on the screen." "There was a lot of things to take in, I had to do a lot of searching." But some subjects were able to digest the information better than others. Those who were able to make sense of the menu layout agreed that having all the information on the screen at once gave them a holistic picture of the design process and the menu layout helped them to understand the metal-ceramic brazing design process. "It was difficult to understand the layout at first but eventually I figured out I can constrain and the software optimizes everything else for me."

The problem with the menu layout seemed to be that of organization. Most of the users thought the menus were randomly placed on the screen and would like the menus to be categorized according to function. A few subjects suggested that there be a "Materials" category for the "Metal", "Ceramic", and "Filler" menus, a "Operations" category for "Braze Technique", "Braze Atmosphere", and "Braze Temperature", and a "Results" category for "Cost", "Braze Strength", and "Braze Stress." Some preferred to have a more layered layout with the primary design parameters like "Metal", "Ceramic", and "Filler" at the top layer and the other secondary parameters at lower layers depending upon their importance. Other subjects suggested a left-to-right arrangement to give a sense of procedure, with the materials on the left side, the operations in the center, both for the user to set, and the results on the right for the user to collect as data. A more organized menu scheme would help the users to better understand the metal-ceramic brazing procedure both because of the visual logical flow, and because of the reduced clutter on the screen which would enable the user to more easily digest the on-screen information.

Another contributor to the clutter of the screen was the inability of the toolkit and PHIGS to make use of the entire screen. About 20% of the screen was not utilized because of this limitation. This was a major restriction to the implementation of the interface and lead to problems like clutter and small text size which were some of the most commonly encountered problems during the test sessions.

An unintended source of distraction was the highlighting of the menu items. The menus were originally designed without highlighting in mind, but highlighting was a default feature of the toolkit used to implement the interface and was not turned-off when implemented. This proved to be a major source of confusion as the subjects interpreted the highlighted menu item as having special importance where in actuality, the highlight only signified the last menu item selected. The highlighting also distorted the appearance of the push button and many subjects had difficulties distinguishing between selected and unselected push buttons.

The purpose of this project was not to study user interaction with geometric models, but most of the subjects commented that they liked the combination of geometric models and design parameters in one layout. However, it was also noted that the geometric models and design parameters were perceived by the subjects to be detached because the geometric models did not seem to respond to all the activities what were happening with the menus.

An important part of the Metal-Ceramic Brazing and the main CAD screen was the button bar at the bottom of the screen because it held the buttons of the major functions for each screen. However, this button bar often went initially unnoticed by most of the subjects because it had to compete with a screen full of buttons and menus. Hence, the subjects were sometimes confused about what to do next. For instance, after selecting the "Metal Part" and the "Ceramic Part", and not noticing the "Start Application" on the button bar, one subject asked: "What do I do after I have selected them [the "Metal Part" and the "Ceramic Part"]?" One solution to this problem would be to change the color of this button bar to red both to make it stand out from the generally blue layout and to emphasize its importance in each screen.

Wording also played a big part in the subjects' understanding of the menus. The "Optimize" and the "Constrain" buttons were intended to be an indicator of parameter status. That is, if the "Optimize" button was selected for a particular design parameter, then that design parameter would always be optimized by the system. However, this concept eluded some of the subjects. They understood the "Optimize" button to be an action that they ask the system to perform, so the subjects would select the button every time they wanted a parameter optimized. "What do optimize mean?" "Optimize is too broad, what am I optimizing?" The subjects' confusion stemmed from the fact that optimize and constrain are verbs which indicate action instead of adjectives to indicate status.

Other wording problems came from design parameter titles. For example, there was confusion regarding the difference of "Braze Stress" and "Braze Strength" since they were both pressure units. "Stress in Joint" and "Strength of Braze" might have been better descriptions of the parameters.

5.3.2 Visual Cues and Feedback

Visual cues and feedback played an important role in the interface, both in keeping the user informed about design specifications, and in helping the user understand metal-ceramic brazing. For instance, the metal-ceramic brazing design process was presented in the interface as a circular system to emphasize the need to iteratively improve the braze design. However, before an iteration could be made, the metal-ceramic brazing process required certain procedures be specified, including the selection of braze surfaces and the assignment of the material to part geometry. Unfortunately, the lack of a clear indication of to what degree these procedures had to be completed seemed to be a great concern for most of the subjects tested.

For example, only two subjects remembered to select the braze surfaces when performing the second set of benchmark tasks which did not explicitly instruct the subjects to select the brazing surface. Instead of warning the user of this omission, the system allowed those subjects who did not select the braze surfaces to continue and iteratively refine the braze design. This apparent breach of procedure was the combination of oversight in the interface design, both by allowing the user to optimize the parameters without first selecting the braze surfaces, and by the lack of verbal cues. A prompting like "Please select braze surfaces" in the verbal cue box at the center of the design parameters when the user first enters the Metal-Ceramic Brazing screen would have been a simple solution to this problem. "Please set design constraints" after the surfaces have been selected would also incur a sense of procedure and would not have left the user to wonder.

As mentioned above in Section 5.3.1, some users had difficulties in grasping the concepts of constrain and optimize. Because this was crucial to the interaction of the interface, clear and timely feedback were needed to convey these concept. The addition of verbal cues, such as "Metal constrained to Steel" or "Metal optimized to Steel" in the verbal cue box after the "Constrain" or "Optimize" buttons of "Metal" have been selected, would have reflected the user's action and would have helped to clarify the meaning of optimize and constrain. The constrain and optimize concepts could be reinforced still further by making the "Optimize" button green when it was selected. This would have helped to promote a clearer sense of exclusivity between the "Optimize" and "Constrain" buttons as well as conveying status--red meaning the parameter was constrained and green meaning the parameter was optimized.

Also mentioned in Section 5.3.1 was the apparent detachment between the geometric models and the design parameters. It was not the focus of this project to study the representation of parameters on a geometric model and therefore no extensive work was done to alleviate this problem. However, the coordination of the color of the "Ceramic" menu and the "Metal" menu along with their respective geometric model would have been a simple and effective means to draw the two together. Another solution would have been to add a label, indicating the material of the geometric model, e.g., "Steel," to the geometric model after the user has designated the brazing surfaces. This would also have served as an additional feedback for when the subjects were selecting the brazing surfaces, the lack of which confused some of the subjects.

An important feature in the metal-ceramic brazing interface was that the system made recommendations to the user based on brazing rules and user input. A clear understanding of these recommendation made subsequent design decisions easier for the user. "Matches:", "Range:", and their respective numeral played the part of conveying the number of matches and the range of recommended values for the design parameter. All the subjects understood these visual cues, however, some subjects showed some hesitation. One subject asked: "Why a range of dollar?", but went on to explain that the range was the range of possible cost and the figure in field was most likely for the selected constraints.

When given an option to choose between different non-numeric options such as materials or braze technique, it was important to indicate the preferred choice. For this purpose, a number was placed on the button of each selection, in the material list pop-up menu, to indicate how the system ranked the selections as well as placing the preferred selection in the field of the parameter's menu. Most of the subjects understood the preferred selection by its being displayed in the field, but did not see the number of recommended rank. To make the ranking of the selections more apparent, the numbers could have been placed on the left side of the each selection button in a distinct column.

One subject commented: "I like to know what's the basis for these options. Where do these rules come from and if they comply with industrial standards." The Metal-Ceramic Brazing interface did allow the user to view all the brazing rules in effect by selecting the "Brazing Rules" button at the bottom of the screen, which this subject did not select. At present, there is no definitive solution for this problem except to change the buttons wording to "Brazing Rules in Effect" which would not fit into the button and would clutter the button so as to draw even less attention.

A feature of the metal-ceramic brazing interface was that it allows the user to select push buttons that are grayed-out. However, the gray text did not stand out on the gray menu which made it extremely difficult for the subjects to read. A simple solution to this problem was to use black text on the push button that were supposed to be grayed-out.

When a grayed-out push button was selected, a pop-up menu would appear to explain why this button was grayed-out. In the case of selecting the materials, the pop-menu indicated the brazing rule with which the selection was not compatible. Although no subjects had difficulties with this, an explanation that included the parameter(s) with which the selection was not compatible along with the non-compatible brazing rule(s) would have provided a more complete explanation.

Upon exiting the Metal-Ceramic Brazing screen, a pop-up menu appeared to give the user the option to "Exit", "Save & Exit", and "Dismiss." Some subjects commented that it would be great if the system could recognized it the model had already been saved and, in that case, just exit without displaying this pop-up menu. However, some subjects liked the pop-up menu and made comments as this one subject did: "I'm a worrier so I would save again before I exit."

"Accept" and "Dismiss" were used as opposed to the more conventional "OK" and "Cancel" in most Microsoft Windows interfaces. Some subjects were a little unsure of their meanings because of the unfamiliarity and because of previous computer experience with Microsoft Windows. However, no subject had difficulty understanding the meanings of "Accept" and "Dismiss." Some subjects even said that they felt "Accept" and "Dismiss" better described the function of their respective button.

5.3.3 Handling Errors

The vendor button in the material selection dialogue box was grayed-out until a material was selected. Many of the subjects did not recognize the vendor button because it was grayed-out and was difficult to read. A better method of interaction would have been to make the vendor button always active, even when no material was selected, so the user could be more aware of this option. Then, if no material was selected, it should ask the user to select a material with a dialogue box containing the available materials.

Many of the subjects instinctively tried to select the geometric models before selecting the "Select Braze Surface" button. When the subjects found that they couldn't just select the surfaces, some became lost as to how to select the surfaces because they didn't see the "Select Braze Surface" button. Besides changing the menu bar's color to red as suggest above, another solution for this problem would have been to display the "Select Surface" dialogue box when the user selected the geometric models without selecting the "Select Braze Surface" button. Since there was no other reason why the user would have selected the geometric model besides selecting the braze surfaces, the system should have recognized that the user's intention in selecting the geometric model was to select the braze surfaces. This would be a more intuitive path of interaction instead of insisting that the user first select the "Select Braze Surface" button and then select the braze surfaces.

5.3.4 Help

Although no on-line help was implemented, most of the subjects liked the fact that a "Help" always available and consistently displayed at the bottom of the screen.

5.3.5 Miscellaneous

Many of the subjects tried to start the metal-ceramic brazing application by double-clicking on the "MC Brazing" button. This method of starting the application was originally designed into the system. The lists of applications, projects, parts, and rules were to be selections in a list field which would allow them to be double-clicked. However, the toolkit utilized to implement this interface did not have a list field tool, therefore menus and push buttons were used to replace the list fields and list field elements. Unlike list field elements, the push button did not accept a double-click and thus, this option was not implemented.

5.3.6 General Impression

One of the goals of this project was to test the subjects' understanding of the application domain and their ability to perform specific given tasks without previous exposure to the application domain or the application. The subjects' responses to the post-benchmark task question, which asked the test subjects of their general impressions of the metal-ceramic brazing interface, clearly indicated that even with minimal domain knowledge, they were able to perform the given tasks with relative ease and high level of satisfaction: "It is good because I don't know metal-ceramic brazing and it makes choices for me. But it still allowed me to change these choices if I wanted to." "Would help if I know more about metal-ceramic brazing. But MC Brazing just seems to be plug and chug." "I think it's pretty cool just by making a few selections I can get something that takes years of study. But I don't know why." "I don't know what is the gap size rule but I know I can't braze the 2 materials." "I can do what I'm asked to do but I don't have a clue." "After doing it once, you can go back and do it pretty easily." "I really enjoyed it, it is a lot of fun." "I like it!"

5.4 Qualitative Data for Redesigned Metal-Ceramic Brazing Interface

Five additional engineering subjects participated in the testing of the redesigned metal-ceramic brazing interface. Although most of the subjects were able to correctly complete the tasks with little or no problems, there were a few that had difficulties with understanding some of the wordings and with the busyness of the menu layout.

Once again, the fact that the interface was a prototype, though refined, brought about some difficulties and probably caused the most frustrations during the test sessions. However, this was explained to the participants before each test session, and they were very understanding and cooperated accordingly.

5.4.1 Layout

The initial impression of all the subjects was that the metal-ceramic brazing screen was too busy. However after completing the tasks, all except one subject thought the screen was not too busy at all. "Busy screen! There's so much stuff here, but it may be convenient once you know how to use it... Not too much information to digest but the way it's arranged makes you feel lost for a while at first." "I'm just reading through all my options here... I had to familiarize myself first, but once that was done, everything's pretty easy. I just had to stare for five minutes and never have to do it again. I think it's fine. No, not too cluttered... It makes a lot more sense if you give the screen a look over." "I like the interface, fairly intuitive. [Kind of] busy though but if you get the layout of it, you can get around... Now that I look at it, it doesn't look confusing at all. It makes perfect sense!"

"It's too much for [a] new user" seemed to be an accurate assessment of the menu layout. One subject asked: "There is a lot here, but what can you do?" Two paths could haven been taken to alleviate this busyness for the novice user: (1) Arrange the menus such that the operations are on the left and the results are on the right, such that the menus can categorized under additional headings. This would further organize the screen and would show how of the design process flows. (2) Divide the design parameters into primary parameters and secondary parameters, and place them on separate layers of menus in accordance with their importance.

Although most of the test subjects had initial difficulties with the menu layout, they were able to correctly complete the benchmark tasks. Given the high cost to change the menu layout and the fact that the existing menu layout did not create major usability problems, the solutions suggested above should be implemented and tested on an experimental basis. The change in the menu layout would be a dramatic change to the interface and may cause more usability problem than it solves.

Another issue that contributed to some of the subjects confusion was the "Constrain" and "Optimize" buttons. "You have 'Constrain' and 'Optimize' buttons everywhere, so makes the screen confused." Besides serving as function buttons, the "Constrain" and "Optimize" buttons also served as status indicators for each design parameter. Consequently, the redundancy of the buttons was necessary for this specific interface design.

Throughout the two iterations of testing, subjects had difficulties spotting functions on the main menu bar located at the bottom of the screen. The color of this menu bar was changed from blue to red, after the first round of testing, so that it would stand out more. But it still presented some difficulties to some users during the second iteration. One subject commented: "Usually the main menu is at the top of the screen." Hence, the main menu might have commanded more attention if it was placed either at the top of the screen beneath the screen title or at the right hand side of the screen.

5.4.2 Visual Cues and Feedback

Overall, the visual cues and feedback were sufficient to lead the subject through the benchmark tasks. Some of the subjects commented on the visual cues and feedbacks: "That's pretty straight forward." "'Save' is pretty obvious here. Big letters." But some of the visual cues and feedback were less than effective in achieving their intended purposes.

The addition of verbal cues in the center verbal cue box received some positive feedback from the subject as a good: "Copper ABA is the best filler because it says 'Optimized to Copper ABA.' Pretty straight forward." "Good to see what you have just done."

However, some of the subjects had a tendency to completely ignore the verbal cue box. When asked whether he noticed the verbal cues in the verbal cue box while performing the benchmark tasks, one subject answered: "Probably not. 95% of the people just read instructions and don't read the screen." Some of the subjects thought the verbal cue box "didn't really standout, maybe [try to make it] brighter." Although the verbal cue box was in the center of the parameters, it had to compete with all the menu and button. Being a dull gray color, it was easily lost in the midst of all the brighter menus and buttons. Some subjects suggested that they just didn't wish to have "another thing to read." The repetitive nature of the verbal cue box also caused it to lose its prominence in the subjects' attention. The subjects simply ignored it after picking up a certain pattern in the displayed messages. "It would be helpful for about the first week, then you would be so familiar with it that you wouldn't need it."

Although the verbal cue box was a good source of feedback and instruction for the user, it could not be depended upon as the sole source of feedback for important tasks such as asking the user to select the brazing surfaces. Other more graphic feedbacks or error handling techniques would be necessary for these purposes.

Most subjects liked the layout combining the geometric model and the design parameters on one screen. However, they also found that the geometry and the parameters "seemed separate", in particular because the geometric model did not respond much to parameter changes. Hence, the usability problem caused by the lack of feedback on the geometric model suggests that the subjects tend to be graphically oriented, and that it is important that proper geometric feedback is provided to reflect parameter changes.

Another feedback problem occurred when participants were looking for compatible ceramic materials. Some users looked for the compatible ceramic materials in the "Metal Material" selection dialogue box. A possible solution would have been to have a "Ceramic" button in the "Metal Material" selection dialogue box, which, in turn, would take the user to the "Ceramic Material" selection dialogue box which would show the ceramic materials that are compatible with the currently selected metal material. Similar buttons and interaction could also be made available for filler, braze method, and braze atmosphere.

The change to make the ranking numbers by the materials in the material selection dialogue box more distinct proved to be fruitful. Some users appeared to understand the numbers to be the recommendation rankings of the materials. "I wonder why the order is 1,2. Maybe it recommends Alumina." The implementation of the "Matches" continued to receive positive feedback from the subjects: "2 Matches, that's convenient. If there's one [match], I wouldn't even bother looking."

One of the subjects did not understand the requested action by the "Select Brazing Surface" dialogue box to "Please select the brazing surface(s) on the Metal part." Only one other subject in the first round of testing had problems with the wording of this sentence. This seemed to be a problem that stemmed from the complexity of the action. The action asked the user to both specify the nature of the geometric part and to select its brazing surface. "How [do] I tell it to select [the] metal one and [the] ceramic one." One possible solution is to default one geometric model as metal and the other as ceramic, and add a button which would toggle the material nature of the parts. This would separate the task of selecting a surface from that of assigning the material nature. Changing the wording of the dialogue box might also solve the problem, but a better worded sentence was elusive.

5.4.3 Handling Errors

As mentioned in the previous section, some subjects did not read the verbal cue box and therefore did not see the "Please select braze surfaces" message. These subjects failed to select the braze surfaces before changing the parameters. One solution would be to bring up the "Select Brazing Surface" dialogue box as soon as the application is started. This would ensure that the user would not forget to select the brazing surfaces.

5.4.4 Miscellaneous

The managing interaction of the projects, parts, rules, and applications in the main CAD screen continued to present problems. "The managing of the projects and files are not intuitive." A more intuitive style of interaction which treats each model as intended would be a major break-through in CAD GUI. Although this is important to all CAD systems, the lack of a paradigm for such interaction has made the development of an intuitive management system very costly.

Another problem that resurfaced in the second iteration of testing was that the font size remained too small.

5.4.5 General Impressions

The following are some of the quotes from the second iteration test subjects about their general impressions of the redesigned metal-ceramic brazing interface: "I think it is convenient and versatile. It took some time to figure out at first but it only took a few minutes." "Instead of hiring engineers, you can hire secretaries to do this." "Knowing a little bit about metal-ceramic-brazing and constrain and optimize first would help like most programs." "Good job, it's impressive." "I'm not sure what the gap size rule is but that's why we can't use Steel. I don't know how everything relates but certain metals can't be used with certain methods." "It's fairly simple, very user friendly. Everything's pretty much right up there. Everything's fairly obvious and pretty easy to use. It's pretty easy for someone, even for me, to come in and make a braze." "The package is easy to use with the user interface, but I still want a manual with examples to follow."

CHAPTER 6. ANALYSIS

The scarcity of usability studies on CAD systems has prompted three questions: (1) Is usability testing different for CAD systems than conventional software? (2) What role does domain expertise play in the use of CAD software since most CAD applications are not simple tasks? (3) How does the design parameters' presentation affect the user's understanding of the design, the design domain, and the capabilities of the CAD system? This chapter presents observations to these questions made during this study.

6.1 Usability Testing in CAD

It has been the goal of this thesis to demonstrate that HCI methodologies (i.e., formative evaluation and iterative refinement) can be applied to CAD system development to introduce a systematic process for improving the usability of CAD user interfaces. CAD user interfaces, which generally are more focused on the application rather than the interface, have been more oriented toward functionality than usability. Although most CAD interface designers would comment that they follow HCI guidelines in their designs, there have been neither studies in the CAD community to show the effects of incorporating HCI guidelines into the CAD interface design nor has there been a process to improve the usability of CAD interface design. No studies which evaluated the usability of CAD systems or, more importantly, proposed solutions to usability problems with existing CAD packages, were found. Most books on CAD recognized the importance of human factors in the design of the graphical user interface. However, the discussion often digressed into a listing of basic HCI interface design guidelines or a description of existing CAD interfaces. Thus, it has generally been stressed that it suffices to be sensitive to usability issues in design, rather than achieving a usable interface as a result of a usability oriented design process.

An iterative refinement cycle was completed for the development of the metal-ceramic brazing interface, though not without difficulties. Some of the difficulties were inherent to the iterative refinement process while others were specific to the CAD system. Though not a unique characteristic of CAD user interfaces, one of the main difficulties with CAD systems in general appears to be their size and complexity.

The I-DEAS interface evaluation performed for this thesis asked the subjects to perform a finite-element analysis through a basic and direct route. Even so, the benchmark task set was extensive and very complicated for the subjects to follow. Typically, the test sessions lasted from two to three hours and most sessions were terminated before the subject completed the tasks. The scope and extent of these sessions were purposely limited to focus on getting an indication to whether an iterative refinement approach with a formative evaluation could be applicable to a CAD system. As demonstrated by the development of the metal-ceramic brazing user interface, this approach can be applied to develop CAD systems.

Nevertheless, a few issues about the complexity of CAD systems were apparent from this study. First, the complexity of the CAD systems can in some instances be reduced: It was observed that some of the existing structures can be simplified without altering the system's functionality. For instance, the I-DEAS FEA module requires users to always create solution sets and FE models. These are not part of the finite-element analysis but are the means by which I-DEAS allow users to manage solutions and parts for finite-element analysis. The system could have been designed such that the user only need to manage the solutions and parts when the next step is not obvious (e.g., when there are multiple solutions or parts).

Requiring certain steps from the user does not necessarily constitute a bad design. However, it is a poor design if the system demands more steps from a user than what is really necessary. For example, existing finite-element packages are generally structured with three distinct modules: (1) the data entry module, (2) the solver, and (3) the postprocessor. In the case of I-DEAS, the user is required to go into each individual module and complete a set of tasks before going onto the next module. These are steps required by the finite-element method to perform an analysis, but they are not necessarily in the order used by the user when mentally performing an analysis.

The second problem with the complexity of CAD systems is that an obvious organization of all the functions is difficult. For instance, when commenting upon the organization of the metal-ceramic brazing user interface, one test subject said: "The screen is too cluttered and seems to lack organization." Most subjects had the tendency of looking through menus trying to familiarize themselves with the system by knowing and understanding the functions that the system offered. The interface's failure to clearly convey the organization of the system made understanding the system even more difficult. In addition to not helping the user to function desired functions, the system's apparent disorganization also contributed to users' confusion about the requirements of the system and the requirements of the application domain.

The problems mentioned above are frequently encountered in complex systems and are usually attributed to the lack of task analysis of the systems' functions. In complex systems, especially for CAD, there is a tendency to develop functions that gives the user access to the latest technology or functionalities that the developers believes would be helpful in certain situations. However, if these functions are not organized according to tasks, they will not be helpful as intended, but become distractions and hence will be ignored. For example, the icons in the I-DEAS interface were recognized by all subjects as being a good interaction style, but most did not understand what the icons meant and hence simply ignored them. This illustrates that by offering a multitude of choices, I-DEAS impressed the subjects as being very powerful. At the same time, the subjects were also lost in the multitude of choices, in part because the application and its domain were new to them. Complexity is sometimes necessary in CAD systems like I-DEAS. But, the large amount of functions in a CAD system should not be a restriction to the user's ability to perform tasks. Instead the functions should be focused on the most performed tasks of the system, and thereby greatly assist in enhancing the system's learnability and usability.

Aside from complexity, an issue that sets usability testing for CAD systems and conventional software apart is the user's reliance upon manuals. Similarly to most CAD systems, I-DEAS depends on its users to learn the system by working through a set of workshops introducing them to some of the system functionalities. This reliance upon a manual seems to suggest (1) the absence of a usability oriented development process on the part of the CAD system developer, (2) the CAD users' acceptance of the manual as part of the CAD user interface, and (3) the inherent complexity of the system and the tasks at hand forces the need for some minimal level of training; training the subject in this study were never afforded.

Manuals, in general, serve as a good source of reference when difficult situations arise during the use of a software package. However, manuals are usually the last source of information to which a user refers. On the other hand, CAD systems rely on manuals as the main instructional guide for the system. Everything about the system, including educating the users, is placed in the CAD manuals which make them required reading. It is considerably easier and less costly to include a workshop in the manual centered about a certain function, than it is to design the interface around this function. However, this places the burden of learning the system upon the user rather than upon the system developer to design a more usable and learnable interface. This dependence may be a good solution for CAD user interfaces because a more usable interface is rarely inexpensive. However, without a cost/benefit study, it is difficult to make a definitive statement about the CAD systems' dependence upon manuals. In either case, CAD developers need to be sensitive to this reliance and not sacrifice long-term major usability problems for short-term low-cost solutions.

Engineers, the actual users of CAD systems, are generally trained to be tenacious solution seekers, and the engineers that participated in this study demonstrated this quality in their persistent effort to complete the benchmark tasks. This in addition to the CAD system's reliance upon manual seemed to breed an attitude in the subjects that their proficiency in the system was strictly a function of their exposure to and familiarity with the system rather than a combination of time and the usability of the interface design. Because everything about the system was included in the manuals, it appeared to the test subjects, that it was a matter of looking through the manual for the appropriate menu selections. "I accept the fact that there is a learning curve... I would expect a person to skim through the manuals and figure out what it's all about." It seemed to be an accepted characteristic of the CAD system that it should be difficult to learn, and that the only remedy is the manual and long hours with the system.

This acceptance among test subjects offers one explanation of the CAD systems' reliance upon manual. In addition, this reliance also suggests the user's lack of choice in the CAD system they use. Although usability in general determines which software packages a user selects, this does not seem to apply to the choice of a CAD system. Instead, CAD users have to accept reading the manuals because most CAD users do not choose the CAD system they use, rather they are given a system to use.

It should also be noted that although CAD manuals often take users through workshops to complete some imaginary projects, the purpose of these workshops appear to be mostly instructional. For instance, performing some of the workshops in the I-DEAS manual does give insights into the purpose of system, but the closeness with which these workshops simulate real world situations and the purpose of the system, remains unstudied.

6.2 Domain Expertise

One of the basic beliefs in HCI is that everyone is an expert and can design a user interface [Hartson94]. A test subject commented that "it's the expert who knows what's intuitive." This comment, along with the fact that most of the subjects expressed the sentiment that they wished they knew more about metal-ceramic brazing before testing the interface, seems to suggest that domain expertise plays a significant role in CAD systems. This is a problem since obviously, not everyone is necessarily a domain expert.

There seemed to be three distinct sets of knowledge that a user may possess to varying degrees when using a CAD system: task knowledge, domain knowledge, and application knowledge. Task knowledge is what the user knows about what is to be performed on the system. CAD users are usually expected to have a sufficient level of task knowledge when using a CAD system since the main use of CAD systems is to perform tasks. The benchmark tasks provided the task knowledge for each subject during the test sessions. Therefore, it was assumed that every subject had a high, if not complete, level of task knowledge.

Domain knowledge is what the user knows about the application domain. The level of domain knowledge gives an indication of the experience and the academic knowledge the user has in the technical background of the tasks. Finite-element analysis and metal-ceramic brazing were the application domains. In the demographics form, most subjects indicated that they had little or no knowledge in either application domains (Section 5.1).

Application knowledge is what the user knows about the system itself. This serves as an indication of the user's pervious experience with the specific software. The I-DEAS finite-element module and the metal-ceramic brazing interfaces were the applications. The subjects, with the exception of one who had previous experience in I-DEAS, had no experience in either application.

The metal-ceramic brazing interface was designed such that only a minimal domain knowledge is required by a user. In fact, even after using the interface, the subjects did not understand why certain materials are compatible with other materials. However, they did pick up that only certain materials could be used with certain other materials in a braze. This illustrates how the design parameter recommendations of the system allowed users to perform the benchmark tasks without having to understand the significance of every design parameter and how each parameter affected the other parameters. The benchmark task simply asked the subjects to specify a certain number of design parameters, after which, the system was able to fill in the blanks for the user. The subjects were not required to have a high level of domain knowledge to perform the benchmark tasks. Most subjects seemed to find satisfaction in the fact that they were able to perform a task without having to understand all the technical background behind the task. "I think it's pretty cool just by making a few selections I can get something that takes years of study. But I don't know why." "I don't know what is the gap size rule but I know I can't braze the 2 materials." "I can do what I'm asked to do but I don't have a clue."

On the other hand, I-DEAS seems to require that users to have a high level of domain and application knowledge in order to perform the benchmark tasks. Specifically, users have to be familiar with the ordered steps of finite-element analysis (i.e., specify the boundary conditions, solve the problem, and display the solution) as well as working with mesh elements and nodes, in order to perform an analysis.

In addition to requiring a high level of domain knowledge, I-DEAS also seems to require a high level of application knowledge as there are a preset number of steps that must be taken to perform the benchmark tasks. Specifically, I-DEAS requires users to create an FE model before starting the analysis. Users must also create a solution set, specify the nodes to be displayed, pick individual nodes for a boundary condition, specify the results to be displayed, select the elements of which the results are to be displayed, and navigate through the list of "Tasks" to find all these functions. If a user performs these steps out of order, then the system will not allow the user to go on to the next step. This often left the subjects confused as to what to do next since the system did not allow them to go on to finish the task. For instance, if they inadvertently made a mistake, the system would not allow them to back track easily to correct their mistake. Hence, many subjects would exit I-DEAS and restart because they did not know how to rectify their mistakes. It was therefore the responsibility of the user to know, and to know well, the sequence of tasks required to perform the analysis.

The results of the test sessions seemed to indicate that the less domain knowledge required by the system, the easier it was for the user to learn the application. Hence, the emphasis of a more learnable interface should not necessarily be to instruct users in the technical background of the tasks, but rather to allow users to quickly understand the workings of the application.

The system developers should be sensitive to the fact that users are sometimes not domain experts. For instance, FEA is frequently used to approximate stresses, a task which is only practical if performed with computers. This may put some otherwise very knowledgeable engineers in the place of a domain novice. However, the user's limited knowledge of the specifics of FEA should ideally not hinder the user in performing this task, rather the system should aid the user in performing this task. The separating of domain knowledge and application knowledge would allow system developers to see more clearly what is required to support a task and what domain knowledge demands are placed on the user. This distinction could also aid CAD system designers in being more accommodating to domain experts while still supporting domain novices.

The results from this evaluation regarding domain knowledge might suggest that a more usable and learnable interface encourages users to learn about the application domain. For example, some of the subjects indicated that they wish they knew more about metal-ceramic brazing after successfully completing the benchmark tasks for the interface.

Another possible benefit of separating domain knowledge from application knowledge would be that it might help a domain expert learn the application faster. Some of the subjects indicated that they had a basic understanding about FEA. Nevertheless, these same subjects struggled with the I-DEAS interface. For example, one participant commented: "I may understand the concepts but I don't know how to accomplish anything." It was even suggested by some subjects that it might have been easier for them to write a program which would calculate the solutions instead of learning the I-

DEAS interface. A more task-oriented development approach as suggested in the previous Section, along with knowledge separation, would be a possible path toward enhancing the usability of a CAD system.

6.3 Presentation of Design Parameters

One of the subjects commented that "scientists have a tendency of communicating in a way such that only scientists can understand them." Organization of functions, and in the case of the metal-ceramic brazing interface, design parameters was extremely important because it communicated direction and purpose to the user. The subjects relied on the system to lead them through the design process because they were novices in both the application and its domain. Hence they tended to be at the mercy of the interface's organization to help them perform the benchmark tasks. For instance, I-DEAS did not cue the subjects towards the next required step; hence they often were lost and, in some cases, were unable to recover to complete the task. Even in the metal-ceramic brazing interface, most subjects failed to select the brazing surfaces when the benchmark tasks has not specifically instructed them to do so. Interface organization was important as the subjects tended to organize their thought process in accordance with the interface.

The forward presentation of the design parameters tried to impressed upon the users the metal-ceramic brazing process. This was done to teach users of the metal-ceramic process and because of the relatively fewer graphical components to the metal-ceramic brazing process. Most of the test subjects commented they were able to understand the metal-ceramic brazing process through their interactions with the design parameters. However, the lack of geometric response from the parameter selections made the geometric model and the design parameters seem unrelated. Future work should devote more attention to this relationship between the geometric model and the design parameters as it presented itself to be of major importance.

The metal-ceramic brazing interface placed the burden of recognizing its organization upon its users. The original goal of the design parameter layout was to convey a wheel and hub relationship between the parameters: all parameters affect all other parameters through those in the center. However, this relationship seemed to be of little importance when the subjects were performing the benchmark tasks. Instead, parameters such as the materials ("Metal" and "Ceramic"), the filler, and the cost of the braze were the most important ones because they essentially formed the braze, and because they were the parameters that were most often manipulated by the users. Therefore, the parameter layout appeared to be disorganized to the subjects. Nevertheless, the layout allowed the subjects to play around with the parameters and most of subjects understood the "Optimize" and "Constrain" concepts well enough that the benchmark task did not present any apparent problems.

Some of the subjects commented that a left to right orientation of the design parameters would stress the flow of the design process while some preferred a more categorized layout of the design parameters. This suggests that the subjects acquired their understanding of the domain knowledge through the layout and their interaction with the design parameters. Most of the subjects answered that they understood more about metal-ceramic brazing after evaluating the interface, and in particular that it is a process of matching compatible design parameters. The organization of the parameters seemed to help the subjects understand the fundamental steps of the process without comprehending the domain knowledge. That is, even though the subjects understood the concept of matching compatible design parameters, they had little idea as to why certain parameters are compatible to each other. This seemed to facilitate the user's ability to perform specific tasks without burdening them with a large amount of domain knowledge. Thus the importance in the metal-ceramic brazing design parameter layout seemed to be that of function instead of relations. The subjects were more interested in how to complete a task rather than the nuances of why and how the parameters were related.

Although this does not directly address the issue of design parameter presentation, the peril of CAD interfaces commanding more confidence in the users than they merit should be noted. For example, the I-DEAS FEA module gives very specific numbers indicating the stresses that occur in a part. This can lead users to believe that the system is highly accurate in its analysis but in fact, FEA is most often only an approximation. Even the apparent expertise of the metal-ceramic brazing interface is a work still in the early stages of development and its results should be checked by the user. However, these interfaces do not warn the user of this apparent mistrust. This places an additional burden on the novice user to gain domain knowledge in order not to be falsely led by the system.

CHAPTER 7 CONCLUSION

This work was the first to apply HCI methodologies to the development of CAD software. It introduced CAD and HCI to each other while exploring three specific questions concerning the CAD user interface: (1) Is usability testing different for CAD systems than conventional software? (2) What role does domain expertise play in the use of CAD software since most CAD applications are not simple tasks? (3) How does the design parameters' presentation affect the user's understanding of the design, the design domain, and the capabilities of the CAD system? It furthermore established the need to investigate how a mutually beneficial relationship between the two disciplines can best be realized.

7.1 Conclusions and Contributions

The main purpose of this work was to introduce HCI methodologies to the development of CAD user interfaces. The results from this work point to a promising yet relatively unknown future. Following conventional HCI guidelines, formative evaluation and iterative refinement were used in the development of the metal-ceramic brazing user interface. In addition, a formative evaluation of the I-DEAS finite-element interface was performed.

This work demonstrated that HCI methodologies can be effectively applied to CAD user interface development, resulting in a usability oriented process for developing usable interfaces. However, the complexity of CAD systems and their task analysis made the application of HCI methodologies to CAD user interface development difficult to implement. Other issues such as the dependency on manuals and the lack of users' choice of the CAD system they use also separate usability testing for CAD systems from that of conventional software.

It was also observed that three distinct sets of knowledge were involved in the use of a CAD system: task knowledge, domain knowledge, and application knowledge. By separating and distinguishing the needs of these sets of knowledge, the CAD system developer will be better able to design highly usable interfaces which would better support task performance while educating the novice domain user of the technical background of the application. Hence, a good CAD user interface should adequately support domain and application experts while providing sufficient guidance for the novice.

The organization of the system functions or, in the case of the metal-ceramic brazing interface, the design parameters, was crucial to the subjects' understanding of the application. The untrained subjects who participated in this study relied greatly upon the organization of the interface to lead them through the benchmark tasks because they were inexperienced in both the application and the application domain. In fact, the interface was the only source of guidance the subjects had on how to perform the benchmark tasks. It was also observed that the subjects had the tendency to align their thought process to that of the interface in their attempts to understand the system. In light of these observations, a clearly and intuitively organized interface could provide users with an invaluable insight into the functionality of the system and facilitate their learning of the system.

7.2 Future Work

This section presents direction to some of the most immediate work that can be undertaken to complete the integration of HCI into CAD. These suggestions do not deal with specific usability questions such as those presented in Chapter 5. Instead, they deal with the obstacles of fully merging HCI methodologies into CAD software development.

A cost/benefit analysis would probably be the most useful study at this point. It would determine if the costly application of HCI methodologies to CAD system development can be justified. For instance, it remains to be seen whether the downstream benefits of a more usable interface would outweigh the high cost of evaluating and iteratively refining complex CAD systems. A cost/benefit analysis may also identify which usability problems would benefit the most from moderate use of formative evaluation and iterative refinement.

Task analysis and studies into the progression of design steps, function needs, and the role of task, domain, and application knowledge is essential to improving the usability of CAD systems. As patterns emerge, a standard CAD systems paradigm could be developed. Such a standardization of CAD systems would eliminate much of the fear in novice CAD users since their inexperience would be considered by the CAD system designers. Retraining time would also be reduced when users switch from one system to another. This standardization of CAD interfaces would also provide a paradigm to encourage structured design thought patterns, and hence, would probably also reduce wasted time spent pondering dead ends. This could potentially reduce the complexity of the CAD systems by streamlining the design process and removing unnecessary steps.

Finally, there is a need for studies into the role manuals play in a CAD interface. This would provide better insights into the education of novice CAD users. For instance, exploring alternative educational methods, perhaps by incorporating the educational process into the interface itself, could reduce the learning time of CAD systems while minimizing the fear of interacting with complex CAD systems, a fear often held by novice users.

CHAPTER 8. REFERENCES

[Aaron84] Aaron, M., "Corporate Identity for Iconic Interface Design: The Graphics Design Perspective," *IEEE Computer Graphics & Applications*, Vol. 4, No. 12, December 1984, pp. 123-43.

[Ahlberg94] Ahlberg, Christopher, and Ben Schneiderman, "Visual Information Seeking: Tight Coupling of Dynamic Query Filters with Starfield Displays," in: Adelson, Beth, Susan Dumais, and Judith Olson, eds., *CHI '94*, Boston, MA, April 24-28, 1994, pp. 313-7.

[Akselsen92] Akselsen, O. M., "Review: Advances in Brazing of Ceramics," *Journal of Material Science*, Vol. 27, No. 8, April 1992, pp. 1989-99.

[Albert93] Albert Consulting Group. "User Interaction Evaluation." Saratoga, CA, 1993.

[Alty89] Alty, J. L., and J. Mullin, "Dialogue Specification in the GRADIENT Dialogue System," in: Sutcliffe, A. and L. Macaulay, eds., *People and Computers V*, Cambridge University Press, 1989, pp. 151-68.

[American63] American Welding Society Committee on Brazing and Soldering. *Brazing Manual*, American Welding Society, New York, 1963.

[Bailey93] Bailey, Gregg, "Iterative Methodology and Design Training in Human-Computer Interface Design," *in: INTERCHI '93*, April 1993, pp. 24-29.

[Bass92] Bass, Len, and Joelle Coutaz. *Developing Software for the User Interface*, Addison-Wesley, 1992.

[Bass95] Bass, Henry M., "Automatic Modification of Part Geometry Subject to Manufacturing Constraints," Thesis - Master of Science in Mechanical Engineering, Virginia Polytechnic Institute and State University, 1995 (tentative title - to be completed).

[Bezier72] Bezier, P. *Numerical Control: Mathematics and Applications*, Wiley and Sons, 1972.

[Booth91] Booth, Paul A., "Modelling the User: User-System Errors and Predictive Grammars," in: Weir, George R.S. and James L. Alty, eds., *Human-Computer Interaction and Complex Systems*, Academic Press, 1991, pp. 123-46.

[Bowes75] Bowes, William H., and Leslie T. Russell. *Stress Analysis by the Finite Element Method for Practicing Engineers.* Lexington, Massachusetts, Lexington Books, 1975.

[Brown89] Brown, Judith R., and Steve Cunningham. *Programming the User Interface, Principles and Examples*, John Wiley & Sons, New York, 1989.

[Brown93] Brown, Marc H., and Manojit Sarkar, "Browsing Graphs Using a Fisheye View," in: Ashlund, Stacey, Kevin Mullet, Austin Henderson, Erik Hollnagel, and Ted White, eds., *INTERCHI '93*, Amsterdam, pg. 516, 24-29 April 1993.

[Card80] Card, S. K., T. P. Moran, and A. Newell, "The Keystroke-Level Model for User Performance Time with Interactive Systems," in: *Communications of the ACM*, Vol. 23, No. 7, July 1980, pp 396-410.

[Carroll86] Carroll, Ann Marie, and Barbara Stanley, "Design of Effective Graphic Presentations for User Interfaces," in: *Human-Computer Interaction,* Lawrence Erlbaum Associates, Vol. 2, No.2, 1986, pp. 24-51.

[Davies80] Davies, Alan J. *The Finite Element Method.* Oxford, Clarendon Press, 1980.

[Dewan90] Dewan, Prasun, "A Tour of the Suite User Interface Software," *Proceedings of the ACM SIGGRAPH*, October 3-5, 1990, pp. 57-65.

[Douglas90] Douglas, Sarah, Eckehard Doerry, and David Novick, "QUICK: A User-Interface Design Kit for Non-Programmers," *Proceedings of the ACM SIGGRAPH*, October 3-5, 1990, pp. 47-56.

[Ellis83] Ellis, Andrew E., "An Advanced User Interface for The Layout Phase of Design," Master's thesis, MIT, Cambridge, MA, 1983.

[Ericsson84] Ericsson, Anders K. *Protocol Analysis: Verbal Reports as Data,* Cambridge, Massachusetts, MIT Press, 1984.

[Fenner75] Fenner, Roger T. *Finite Element Methods for Engineers.* New York, Macmillan Press, 1975.

[Fischer89] Fischer, Gerhard, "Human-Computer Interaction Software, Lessons Learned, Challenges Ahead," *IEEE Software,* Vol. 6, No. 1, January 1989, pp. 44-52.

[Fischer92] Fischer, Gerhard, Jonathan Grudin, Andreas Lemke, Raymond McCall, Jonathan Ostwald, Brent Reeves, and Frank Shipman, "Supporting Indirect Collaborative Design with Integrated Knowledge-Based Design Environments," in: *Human-Computer Interaction,* Vol. 7, pp. 281-314, 1992.

[Foley84] Foley, J. D., V. L. Wallace, and P. Chan, "The Human Factors of Computer Graphics Interaction Techniques," *IEEE Computer Graphics & Applications,* Vol. 4, No. 11, November 1984, pp. 180-91.

[Furnas94] Furnas, George, and Jeff Zacks, "Multitrees: Enriching and Reusing Hierarchical Structure," in: Adelson, Beth, Susan Dumais, and Judith Olson, eds., *CHI '94*, Boston, MA, April 24-28, 1994, pp. 330-336.

[Gantt92] Gantt, Michelle, and Bonnie A. Nardi, "Gardeners and Gurus: Patterns of Cooperation Among CAD Users," in: Bausersfeld, Penny, John Bennett, and Gene Lynch, eds., *CHI '92*, Monterey, CA, May 3-7,1992, pp. 107-117.

[Gould85] Gould, J., and C. Lewis, "Designing for Usability: Key Principles and What Designers Think," *Communications of the ACM*, Vol. 28, No. 3, March 1985, pp. 300-311.

[Grant91] Grant, Simon, and Terry Mayes, "Cognitive Task Analysis?" in: Weir, George R.S. and James L. Alty, eds. *Human-Computer Interaction and Complex Systems,* Academic Press, 1991, pp. 147-168.

[Halter85] Halter, R., "Man-Machine Interface Design Challenges," in: *Design News*, Vol. 41, No. 16, August 1985, pp. 63-70.

[Hancock89] Hancock, P.A., and M.H. Chignell. *Intelligent Interfaces,* North-Holland, 1989.

[Hartson94] Hartson, H. R., "Introduction to Human-Computer Interaction," Course Notes for Computer Science 5714, Virginia Polytechnic Institute and State University, Fall 1994.

[Helgesson68] Helgesson, Claes. *Ceramic-to-Metal Bonding*, Boston Technical Publishers, Cambridge, MA, 1968.

[Hicks91] Hicks, Richard, and James Essinger. *Making computers more human: Designing for human-computer interaction,* Elsevier, 1991.

[Hix93] Hix, Deborah, and H. Rex Hartson. *Developing User Interfaces: Ensuring Usability Through Product & Process*, New York, John Wiley & Sons, 1993.

[Hutt93] Hutt, Andrew. *User Interface,* New York, Prentice Hall, 1993.

[John87] John, P., "Ergonomics and Mechanical Engineering Design," in: *Megaw, E. D., ed., Contemporary Ergonomics 87*, Swansea, Wales, April 6-10, 1987, pp. 269-272.

[Jones91] Jones, Rachel, Linda Candy, and Ernest Edmonds, "Knowledge-based System Requirements," in: Bullinger, H. J., ed., *Human Aspects in Computing: Design and Use on Interactive Systems and Information Management*, Elsevier Science Publisher, New York, 1991, pp. 796-800.

[Kiem92] Kiem, Daniel A., and Vincent Lum, "GRADI: A Graphical Database Interface for a Multimedia DBMS," in: *Cooper, Richard, ed., Interfaces to Database Systems*, Springer-Verlag, 1992, pp. 95-112.

[Lee90] Lee, E., "User-Interface Development Tools," *IEEE Software,* Vol. 7, No. 3, May 1990, pp. 31-36.

[Majchrzak87] Majchrzak, Ann, Tien-Chien Chang, Woodrow Barfield, Ray Eberts, and Gavriel Salvendy. *Human Aspects of Computer-Aided Design,* New York, Taylor and Francis, 1987.

[Meyer92] Meyer, Bernd, "Beyond Icons, Towards New Metaphors for Visual Query Languages for Spatial Information Systems," in: Cooper, Richard, ed., *Interfaces to Database Systems*, New York, Springer-Verlag, 1992, pp. 113-125.

[Miller91] Miller, James R., Joseph W. Sullivan, and Sherman W. Tyler, "Introduction," in: Sullivan, Joseph W., and Sherman W. Tyler, eds., *Intelligent User Interfaces,* ACM Press, 1991, pp. 1-10.

[Myers89] Myers, B., "User-Interface Tools: Introduction and Survey," *IEEE Software,* Vol. 6, No.1, January 1989, pp. 15-23.

[Myers92] Myers, B. A. and M. B. Rosson, "Survey on User Interface Programming," in: Bausersfeld, Penny, John Bennett, and Gene Lynch, eds., *CHI '92*, Monterey, CA, May 3-7,1992, pp. 195-202.

[Norman86] Norman, D. A., and S. W. Draper. *User Centered System Design-New Perspectives on Human-Computer Interaction*, Lawrence

Erlbaum Associates, 1986.

[Pattee68] Pattee, H. E. *Joining Ceramics and Graphite to other Materials,* Office of Technology Utilization, National Aeronautics and Space Administration, Washington D. C., 1968.

[Payne86] Payne, S. J., and T. R. G. Green, "Task-Action Grammars: A Model of the Mental Representation of Task Languages," in: *Human-Computer Interaction,* Lawrence Erlbaum Associates, Vol. 2, No. 2, 1986, pp. 93-133.

[Pittman90] Pittman, Jon H., and Christopher J. Kitrick, "VUIMS: A Visual User Interface Management System," *Proceedings of the ACM SIGGRAPH*, October 3-5, 1990, pp. 36-46.

[Powrie87] Powrie, S. E., "Software Ergonomics at the Sharp End: Development of a New Interface for a CAD Package," in: Megaw, E. D., ed., Contemporary Ergonomics '87, Swansea, Wales, April 6-10, 1987, pp. 215-220.

[Rao94] Rao, Ramana, and Stuart K. Card, "The Table Lens: Merging Graphical and Symbolic Representations in an Interactive

Focus+Context Visualization for Tabular Information," in: Adelson, Beth, Susan Dumais, and Judith Olson, eds., *CHI '94*, Boston, MA, April 24-28, 1994, pp. 318-22.

[Robertson92] Robertson, David. C., and Thomas J. Allen, "Managing CAD Systems in Mechanical Design Engineering," *IEEE Transactions on Engineering Management*, Vol. 39, No. 1, February 1992, pp. 23-31.

[Robinson90] Robinson, H., "Towards a Sociology of Human-Computer Interaction," in: Luff, P., N. Gilbert, and D. Frohlich, eds., *Computers and Conversation*, Academic Press, London, 1990, pp. 155-128.

[Rogers88] Rogers, Yvonne., "User Requirements for Expert System Explanation: What, Why and When?" in: Jones, D. M. and R. Winder, eds., *People and Computers IV,* Cambridge University Press, New York, 1988, pp. 546-564.

[Sabonnadiere87] Sabonnadiere, Jean-Claude, and Jean-Louis Coulomb. *Finite Element Methods in CAD,* New York, Springer-Verlag, 1987.

[Sakurai82] Sakurai, H. "Solid Model Input Through Orthographic Views," Master's thesis, MIT, Cambridge, MA, 1982.

[Sastry92] Sastry, Lakshmi, "User Interface Management Systems for Engineering Applications," *Computer Graphics Forum,* Vol. 2, No. 2, 1992, pp. 113-129.

[Schiele90] Schiele, Franz, and Thomas Green, "HCI Formalisms and Cognitive Psychology: The Case of Task-Action Grammar," in: Harrison, M. and H. Thimbleby, eds., *Formal Methods in Human-Computer Interaction,* Cambridge University Press, 1990, pp. 9-62.

[Schleich94] Schleich, R., and M. J. Durst, "Beyond WYSIWYG: Display of Hidden Information in Graphics Editors," in: Seidel, Hans-Peter, and Philip J. Willis, eds., *EUROGRAPHICS '94*, Oslo, Norway, September 12-16, 1994, pp. 185-194.

[Schneiderman88] Schneiderman, B., "We Can Design Better User Interfaces: A Review of Human-Computer Interaction Styles," *Ergonomics,* Vol. 31, No. 5, May 1988, pp. 699-710.

[Schwartz90] Schwartz, M., *Ceramic Joining*, ASM International, Materials Park, Ohio, 1990.

[Sinclair87] Sinclair, M. A., "What is a Good CAD Dialogue?" in: Megaw, E. D., ed., *Contemporary Ergonomics 87*, Swansea, Wales, April 6-10, 1987, pp. 221-226.

[Snowberry83] Snowberry, K., Parkinson, S.R., and Sisson, N., "Computer Display Menus," *Erogonomics*, Vol. 26, No. 7, 1983, pp.699-712.

[Spencer85] Spencer, R. H., "Interactive Software Design: The Human Touch," *Computers in Mechanical Engineering,* Vol. 4, No. 18, September 1985, pp. 45-48.

[Stone94] Stone, Maureen C., Ken Fishkin, Eric A. Bier, "The Movable Filter as a User Interface Tool," in: Adelson, Beth, Susan Dumais, and Judith Olson, eds., *CHI '94*, Boston, MA, April 24-28, 1994, pp. 306-312.

[Suga89] Suga, T., and K. Miyazawa, "Metal/Metal and Metal/Ceramic Interfaces Produced by means of a Surface Activation Method at Room Temperature," in: *Metal-Ceramic Interfaces, Proceedings of an International Workshop*, Pergamon Press, New York, 1989, pp. 189-198.

[Swezey83] Swezey, R. W., and E. G. Davis, "A Case Study of Human Factors Guidelines in Computer Graphics," *IEEE Computer Graphics & Applications*, Vol. 3, No. 8, November 1983, pp. 21-30.

[Thatch87] Thatch, Brian R., "A PHIGS Based Interactive Graphical Processor for Spatial Mechanism Analysis and Synthesis," Thesis - Master of Science in Mechanical Engineering, Virginia Polytechnic Institute and State University, 1987.

[Thatch88] Thatch, B. R., and A. Myklebust, "A PHIGS Based Graphics Input Interface for Spatial Mechanism Design," *IEEE Computer Graphics & Applications*, Vol. 8, No. 2, March 1988, pp. 26-38.

[Tong77] Tong, Pin, and John N. Rossettos, *Finite-Element Method*. Cambridge, Massachusetts, MIT Press, 1977.

[Tsang86] Tsang, Yuk-Piu, "Design and Implementation of an Advanced User Interface for Three-Dimensional Conceptual Design on CAD/CAM Systems," Thesis - Master of Science in Mechanical Engineering, Virginia Polytechnic Institute and State University, 1986.

[Ullman90] Ullman, David G., Stephen Wood, and David Craig, "The

Importance of Drawing in the Mechanical Design Process," *Computer & Graphics,* Vol. 4, No. 2, February 1990, pp.263-274.

[van Zuylen91] van Zuylen, Henk J., "Front-ends for Scientific Programs," in: Bullinger, H. J., ed., *Human Aspects in Computing: Design and Use on Interactive Systems and Information Management*, Elsevier Science Publisher, New York, 1991, pp. 813-819.

[Vershel85] Vershel, Mark, "Designing User Interfaces for Engineering Design Systems" *Communications of the ACM*, Vol. 28, No. 3, March 1985, pp. 234-245.

[Warschat91] Warschat, Ing. J., "MOCAD - an Intelligent Front End for Design for Assembly," in: Bullinger, H. J., ed., *Human Aspects in Computing: Design and Use on Interactive Systems and Information Management*, Elsevier Science Publisher, New York, 1991, pp. 830-833.

[Wilson91] Wilson, M., and A. Conway, "Enhanced Interaction Styles for User Interfaces," *IEEE Computer Graphics & Applications*, Vol. 11, No. 2, March 1991, pp. 79-90.

[Woodgate85] Woodgate, H. Sam, "The Use of Graphical Symbolic Commands (Icons) in Application Programs," in: Vandoni, Carlo E., ed., *Eurographics '85*, Nice, France, September 9-13, 1985, pp. 25-37.

[Woyak92] Woyak, Scott, "Interface Framework Reference Guide," Thesis - Master of Science in Mechanical Engineering, Virginia Polytechnic Institute and State University, 1992.

[Woyak93] Woyak, S. A., and A. Myklebust, "A Motif-Like Object-Oriented Interface Framework Using PHIGS," *1st Annual PHIGS Users' Group Conference*, March 21-24, 1993, Orlando, Florida.

[Wu92] Wu, C-H Allen, Tedd S. Hadley, and Daniel D. Gajski, "An Efficient Multi-View Design Model for Real-Time Interactive Synthesis," *IEEE Software*, Vol. 9, No. 3, May 1992, pp. 328-331.

[Zecher94] Zecher, Jack E. *Computer Graphics for CAD/CAM Systems,* Marcel Dekker, New York, 1994.

[Zienkiewicz71] Zienkiewicz, O. C. *The Finite Element Method in Engineering Science.*, New York, McGraw-Hill, 1971.

Vita

Jason Chung-Cheng Moo was born in Taipei, Taiwan on June 13, 1971. He and his family immigrated to the United States on February 25, 1980. After attending eight different high schools, he completed his high school education at Paramus High School, Paramus, New Jersey, in June 1989. He then attended Stevens Institute of Technologies in Hoboken, New Jersey. Four years later, he graduated with a BE degree in Mechanical Engineering and a BA degree in Literature. Moo continued his education at Virginia Tech where he pursued a Master of Science degree in Mechanical Engineering. After graduating in July 1995, he hopes to continue his search for common-sense in leading-edge technology.