INFORMATION TO USERS

This manuscript has been reproduced from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps. Each original is also photographed in one exposure and is included in reduced form at the back of the book.

Photographs included in the original manuscript have been reproduced xerographically in this copy. Higher quality 6" x 9" black and white photographic prints are available for any photographs or illustrations appearing in this copy for an additional charge. Contact UMI directly to order.

UMI

University Microfilms International
A Bell & Howell Information Company
300 North Zeeb Road. Ann Arbor, MI 48106-1346 USA
313/761-4700 800/521-0600

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Order Number 9509612

Cognitive systems engineering: Human-computer interaction design for decision support

Ehrhart, Lee Scott, Ph.D.

George Mason University, 1994

U·M·I
300 N. Zeeb Rd.
Ann Arbor, MI 48106

Cognitive Systems Engineering: Human-Computer Interaction Design for Decision Support

A dissertation submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy at George Mason University

By

Lee Scott Ehrhart
B.A., History, University of Colorado, 1977
M.S., Systems Engineering, 1988

Director: Dr. Andrew P. Sage
Dean, School of Information Technology and Engineering

Summer 1994
George Mason University
Fairfax, Virginia

COGNITIVE SYSTEMS ENGINEERING: HUMAN-COMPUTER INTERACTION DESIGN FOR DECISION SUPPORT

by

Lee Scott Ehrhart
A Dissertation
Submitted to the Graduate Faculty
of
George Mason University
in Partial Fulfillment of
the Requirements for the Degree
of
Doctor of Philosophy
Information Technology

Committee:

Andrew P. Sage — Director

Andriole

Deborah A. Boehm-Davis

Paul E. Lehner

Leonard Adelman

W. Murray Black — Department Chairperson

Andrew P. Sage — Dean, School of Information Technology and Engineering

Date: August 24, 1994

Fall 1994
George Mason University
Fairfax, Virginia

Copyright Notice

Apple® and Macintosh® are registered trademarks of Apple Computer, Inc.

SuperCard® is a registered trademark of Aldus, Inc.

Acknowledgements

Very few students are fortunate to have a mentor during their doctoral work; I had two. Both the chair of my committee, Dr. Andrew P. Sage, and the co-director of my research, Dr. Stephen J. Andriole, actively supported this research and my pursuit of the doctorate. I was also fortunate to have a multi-disciplinary doctoral committee. During the years I worked on this research, Dr. Leonard Adelman, Dr. Deborah Boehm-Davis, and Dr. Paul Lehner generously contributed not only their time, but also their experience and knowledge.

I owe a tremendous debt of gratitude to our sponsors in the C3 Division at Rome Laboratory, particularly Richard T. Slavinski and Robert Flo. In addition, for allowing me inside a very intense rapid systems development effort, thanks go to the members of the FLEX project team: Tom Clark, Earl Labatt, Gerald Ruigrok, Georganne deWalder, Capt. Scott Bourgeois (USAF), 1Lt. Keith Felter (USAF), and the senior advisors from the FLEX Working Group. Thanks also to Capt. Gary Stefanich (USAF), from the APS Program Office, for his help developing the experimental scenarios, and to Col. Larry Simmons for securing subjects and domain experts from the 509th Air Refueling Squadron. Finally, a very special thanks to Lt. Col. Thomas Triscari, Jr. (USAF), who not only helped to bring this research to Rome Laboratory, but also provided valuable insight into the complexities of large-scale systems acquisition.

I also wish to thank my loyal family and friends. My parents, an engineer and a policy analyst, helped to nurture my interests in both engineering and decision-making. My sister, Dr. Jane Scott-Lennox, and her husband, Dr. Richard Lennox, were my best source of critical commentary and creative brainstorming outside my committee. Dr. Robert C. Ehrhart was very supportive and helped in some of the domain modeling based on his 20-year career as an officer in the US Air Force. Finally, thanks to my son, Clayton, who tolerated the perversity of having his mother in school.

Table of Contents

Appendices

List of Tables

Appendix Tables

List of Figures

Appendix Figures

Abstract

COGNITIVE SYSTEMS ENGINEERING:
HUMAN-COMPUTER INTERACTION DESIGN FOR DECISION SUPPORT

Lee Scott Ehrhart, Ph.D.

George Mason University, 1994

Dissertation Director: Dr. Andrew P. Sage

This research outlines a framework for employing cognitive systems engineering principles and practices to enhance the requirements identification and design phases of system development and improve human-computer interaction (HCI) designs for decision support. The framework focuses on the application of cognitive research and technology in developing a more comprehensive understanding, representation, and translation of the decision-maker's cognitive task requirements in human-computer cooperative decision-making. Additional guidance is provided in a series of tables summarizing research from software engineering, decision sciences, cognitive psychology and other related fields. These supports assist the designer in defining a robust set of system requirements and guide design tradeoff decisions.

Following the presentation of the CSE framework, a system design case study in human-computer cooperative decision-making demonstrates the practical implementation of CSE for HCI design. The case study leads the reader through the application of the guidance tables to a "real world" design

problem. The case study also presents empirical and experimental evaluations of the benefits of the design framework in terms of

- improving the ***process*** of creating information presentation and interaction designs and, consequently,
- improving the HCI design ***product*** to increase the functionality of the delivered system and enhance decision-making performance.

Finally, the prototype for a CSE design practitioner's handbook is presented to suggest a format for making the CSE design framework usable in the information systems engineering community.

1. Designing Interaction for Human-Computer Cooperative Decision-Making

1.1 Problems in Human-Computer Interaction (HCI) for Decision-Making

Critical decision-making systems provide the procedural and technical means by which an organization responds to potentially threatening internal or external events. In time-pressured situations, it is often desirable to relegate to machines the immediate decisions that fall within well-defined boundaries. In the case of more complex environments, the range of possible events and appropriate responses may not be sufficiently constrained for acceptable automated or machine-controlled response. In these instances, it is more effective to have human decision-makers cooperate with a computer-based decision support system (DSS) to determine the appropriate response.

Human-computer decision-making performance in critical situations is dramatically affected by the design of the user-computer cooperation (e.g., task allocation, information sharing requirements, etc.) with respect to the environmental characteristics (e.g., complexity, uncertainty, dynamics, level of threat, etc.) and the response requirements (e.g., timing and precision). Woods and Roth (1988) propose that mismatches in the system design involving these factors result in the ineffective use of resources and, in the worst cases, disastrous system errors and failures. They cite several cases where automation degraded rather than improved performance due to user-related design failures such as a lack of support for supervisory control requirements and decision-making strategies, and failures to anticipate the organizational impacts of technological change. The design failures cited by Woods and Roth suggest that the way that human decision-makers are presented and allowed to interact with information in computer-based decision aids plays an integral role in determining the performance integrity of the delivered system.

Human-computer interaction (HCI) comprises the information presentation and interaction routines that define the communication between the human user and the computer-based system. As such, HCI involves more than the display screens and navigational controls of the system interface. Shneiderman (1992) states:

> Human engineering, which was seen as the paint put on the end of a project, is now understood to be the steel frame on which the structure is built (p. iii).

The primary objective of the HCI design in human-computer cooperative decision-making systems is ensuring that the human decision-maker gets the *right information* at the *right time* with the *right level of detail*.

The generally accepted principles which guide design of the interaction between humans and machines are built on a foundation of research which has accrued over more than sixty years. Man-machine system design initially focused on physical and perceptual tasks. As system designers began to address the interaction between human users and computer systems for word processing and data entry/retrieval, the research emphasis shifted from solely examining physical and perceptual tasks to include cognitive tasks. In the past thirty years, advances in computer technology and software design have extended HCI into the realms of problem solving and decision making. Today, the requirement to field decision aiding systems in real-world environments coupled with advances in computer hardware and software have stimulated a re-evaluation of traditional principles and guidelines for HCI design (c.f., Andriole & Adelman, 1989; Brooks, 1988; Carroll & Campbell, 1988; Duffy, 1993; Ehrhart, 1993a; Gardiner & Christie, 1987; Klein *et al*, 1993).

Technological advances in interactive computing over the past three decades present the HCI designer with an extensive array of tools and techniques for constructing the user interface, but often only vague suggestions for the best use of these resources. Research in ergonomics, human factors, and engineering psychology present an applied approach to the engineering of human-machine systems. The cognitive and behavioral sciences incorporate a broad range of disciplines, including psychology (behavioral, perceptual, cognitive, educational, and developmental), linguistics, sociology, anthropology, and even philosophy providing a mixture of viewpoints on the nature of human-computer interaction. The management and organizational sciences contribute not only the managerial insight acquired through designing management information systems, but also expertise in organization theory, communication theory, and decision sciences. Finally, on-going research in the information technology disciplines (information systems engineering, software engineering, computer science, artificial intelligence, and systems engineering) continues to advance knowledge about the engineering of computer-based systems. There remains, however, a gap between the research findings from these individual disciplines and their synthesis for practical application in HCI design for decision support.

The requirement for a multi-disciplinary approach to human-computer interaction design is not new. Consider, for example, Sheridan and Ferrell's (1974) comments twenty years ago:

> The study of man-machine systems is interdisciplinary, to say the least. It represents the intersection of a variety of applied sciences having to do with man's structure, functioning, and behavior. ... The serious student of man-machine systems must

> somehow acquaint himself with a number of the disciplines which intersect to form the field (p. 15).

Similarly, Salvendy (1987) describes human-computer interaction as a complex, three-way interaction of social, cognitive, and ergonomic factors. Both Norman (1986) and Allen (1982) recognized the need for the synthesis of these multi-disciplinary views into a new discipline: *cognitive engineering*. The goal of cognitive engineering is the definition of a set of principles, drawn from the fields of cognitive science and engineering, to guide the designer in matching the user, task, and environmental requirements to available tools and techniques for the design of optimal interaction.

Design of human-computer cooperation in problem solving and decision-making must be "driven by human cognitive processes, not computer technology" (Andriole & Adelman, 1989). Writing from a control systems perspective, Woods and Roth (1988) use the term *cognitive systems* to describe the teaming of a human user and machine to perform problem solving tasks. Their writing and research is directed toward the definition of principles for a new discipline they refer to as *cognitive systems engineering*. Cognitive systems engineering (CSE) applies multi-disciplinary research findings and design experience from the cognitive science and engineering disciplines to the design, evaluation, and construction of systems supporting and improving human task performance. This focus on task performance, as opposed to interaction performance, distinguishes cognitive systems engineering from Norman's (1986) cognitive engineering paradigm.

In direct contrast to the often noted tendency for "technology push" in advanced systems development, CSE emphasis on supporting the needs of the decision-maker represents *requirements-driven* design. The key premise in this concept is the notion that, in addition to the interaction task requirements (ITRs) associated with operating the interface, performance improvement hinges upon identifying a more comprehensive set of human cognitive task requirements (CTRs) and successfully translating those requirements into design concepts. The resulting system should demonstrate consistently high human-computer decision task performance as determined by appropriate measures of performance and effectiveness. To incorporate these concepts requires CSE design methods that guide the matching of user, task, and organizational, situational and environmental requirements to available tools and techniques for the design of human-machine cooperative decision-making.

Bersoff (1984) defines *product integrity* as a measure of the extent the delivered product satisfies the real needs and the cost, schedule and performance expectations of the user. To ensure these expectations are met, the systems engineering approaches to the design and delivery of systems integrate quality assurance and project controls into the system development process (c.f., Rouse, 1991 and Sage, 1992). The traditional systems engineering systems development life cycle (SDLC) model comprises an iterative, multi-step process to guide designers in developing effective systems. The essential steps include:

1. **Problem definition** - ***understanding*** problem dimensions to enable problem structuring (*why* the system is needed);
2. **Requirements identification & modeling** - ***representing*** system response goals to support design specification (*what* is needed);
3. **Design** - ***translating*** requirements into a functional technological solution (*how* to meet identified needs);
4. **Implementation** - realizing the technological solution; and
5. **Operational testing & evaluation** - verifying and validating system performance against requirements goals and design specifications.[1]

---

[1] Note: Each phase of this process involves internal testing and evaluation to verify that the products developed at that phase meet the stated objectives.

Although the implementation of the technical solution often gets blamed for performance failures, the results of several studies of software-intensive systems traced the majority of errors in delivered systems to the *pre*-implementation phases (Boehm, 1975; Thayer, 1975). Thus, the greatest leverage on improving the product integrity in human-computer cooperative decision systems is to be gained by adopting a systematic method for improving the pre-implementation processes and products. This may best be accomplished by obtaining a more comprehensive ***understanding*** of the users, tasks, and operational context, a more accurate ***representation*** of the technical requirements, and a more effective ***translation*** of requirements into HCI designs.

Figure 1.1 decomposes the elements in system designs for effective human-computer cooperative decision support. The model suggests that the quality of the technological support for human-computer cooperative decision-making depends upon both the HCI design and the hardware/software design. Following the model downward, the HCI design is composed of support for the tasks related to the mechanics of machine input and output are developed as the interaction task requirements (ITRs) and the support for the human decision-making tasks represented in the cognitive task requirements (CTRs). As indicated in Figure 1.1, this research focuses on the cognitive aspects of decision-making as they are supported in the HCI design.

Cognitive systems engineering for HCI design requires the designer to define and integrate multiple models of the decision context, task requirements, HCI requirements, and implementation options. The term "model" covers a wide range of representations with an equally wide variety of uses. The *American Heritage Dictionary* (1985) presents several definitions; three seem most appropriate to this discussion. Using their definitions, with italics added by the author, a model is:

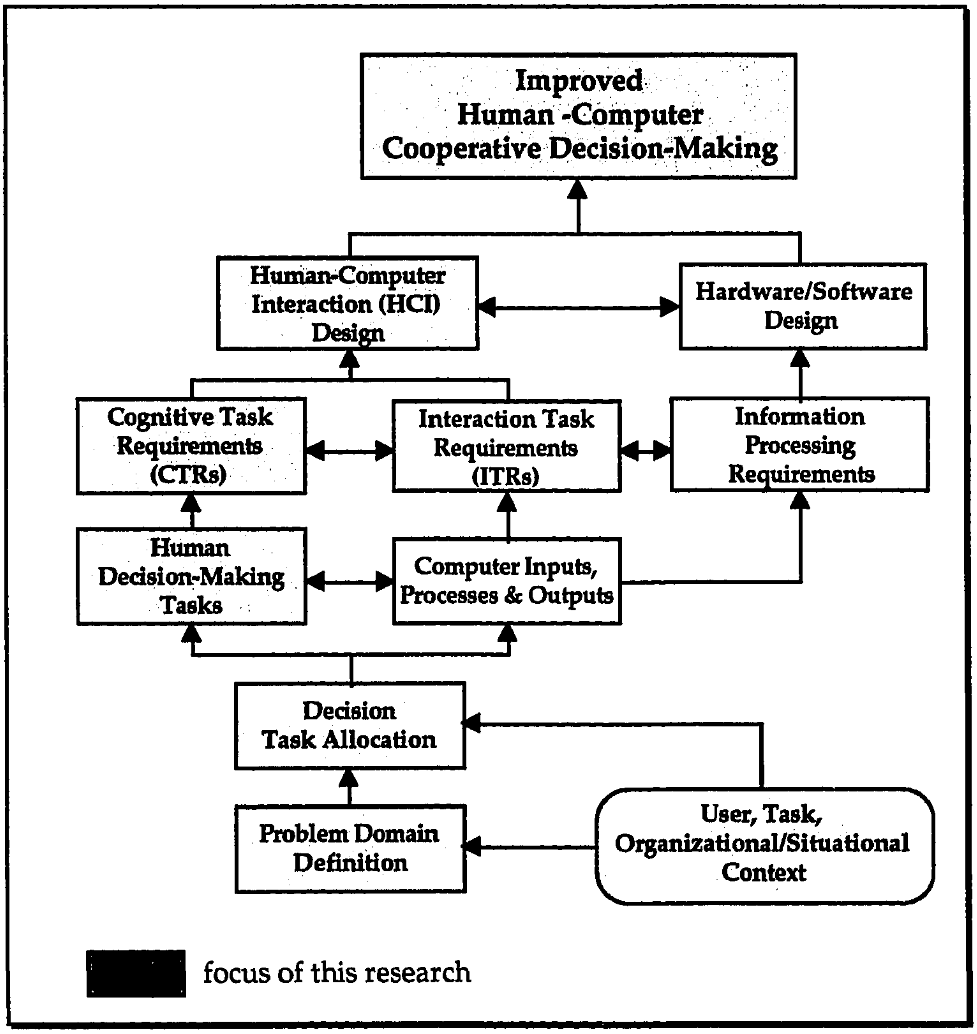


Figure 1.1: Elements of System Development for Effective Human-Computer Cooperative Decision Support

1. "A tentative ***description*** of a system or theory that accounts for all of its known properties."
2. "A preliminary ***pattern*** serving as the plan from which an item not yet constructed will be produced."
3. "A small ***object,*** usually built to scale, that represents another, often larger object." (*American Heritage Dictionary, 1985*, p. 806)

These three definitions encompass the three principal types of models that typically form the context for HCI design: conceptual models, requirements and design models, and developmental (or prototype) models. Design and development are in themselves refining processes in which analysis, modeling, and evaluation interact continually. In the early phases of development, models may be largely informal, conceptual expressions of the designer's view of the system and its context. Evaluation of existing system operations supports the early stages of concept definition that, in turn, form the first system model. Implicit in this model is some representation of the system's purpose as it relates to organizational goals and the identification of criteria by which the achievement of those goals is recognized.

As definition progresses, the current system model is analyzed in terms of the perceived deficiencies, or shortfalls, between what the system provides and what the organization needs. This process leads to the definition of yet another model -- a set of requirements for the next generation system and the criteria by which alternative designs, or system models, will be evaluated with respect to those requirements. Evaluation and modeling continue to play a key role in supporting decisions throughout the iterative process of design. Even in the early phases of development, evaluation is still being performed upon models in the form of system prototypes. Finally, evaluation of operational systems is accomplished with the assumption that the evaluation criteria, established in the form of measures of performance (MOPs) and measures of effectiveness (MOEs), accurately represent (or model) the relationships between component, subsystem, and system performance and the larger purpose for which the system is intended.

Hoeber (1981) identifies three basic purposes for modeling:

- Improve problem understanding for both the analyst/designer and the user/client;
- Assist in developing solutions to complex, yet tractable, problems; and
- Provide support for making choices where uncertainty and ambiguity cannot be resolved or there are no clear-cut solutions.

In each case, the underlying motivation for developing a model is the decision or "problem." Furthermore, there is a implied assumption that the problem is complex enough to be difficult to understand or solve without the aid of a model to abstract the relevant interactions among the various critical aspects of the problem.

Although models are often criticized as simplifying the problem environment, Hoeber suggests that this feature of models is actually desirable in that it is the overwhelming complexity of the real world that limits decision-maker's ability to solve the specific problem at hand. Modeling tradeoffs generally attempt to balance the advantages of greater simplicity against the risks of omitting a potentially critical factor. Table 1.1 presents these tradeoffs in terms of their effect on the credibility of the model and information gained.

There is a tendency in all modeling efforts to model those aspects which are best understood and readily lend themselves to representation. As Table 1.1 indicates, models may vary in degree of abstraction from highly detailed representations of all the tasks, personnel, organizational interactions, automated support systems and environmental factors involved in a decision domain on the one hand to representations of the relationships between a few highly aggregated variables on the other hand. Less abstract models (i.e., those with greater detail) appear more "realistic" and are accorded a high degree of face validity. However, the exhaustive detail may not help the model users to clearly identify the critical factors and interactions. Increasing the number of variables and their associated assumptions also increases the potential for bias in the results. The large number of variables required and the accompanying increase in complexity can result in a model which is unwieldy; thus, it is often infeasible to do extensive sensitivity analysis on all the potentially relevant variables.

Credibility	**Level of Abstraction**	
	Least Abstract	**Most Abstract**
Model	• Appearance of validity due to detail & greater match to real world inputs & processes • Increased complexity may require simulation rather than an analytical model	• Appears less valid due to simplicity & lack of real world detail • Can be supported with historical data or data from more detailed models • Increased potential of omitting important factors
Results	• Large number of variables & assumptions increases possibility of bias. • Large number of variables makes comprehensive sensitivity analysis infeasible. Potentially important variables may not be fully explored.	• Dependent upon the credibility of the model • Simpler model is easier for the client to understand • Simpler model is more tractable for sensitivity analyses

Table 1.1: Effects of Abstraction Level on Model Credibility
(adapted from Battilega & Grange, 1980)

More abstract models used in problem definition and early in the requirements identification phase do not attempt to represent real world inputs and outputs and are less costly to develop. While extraction and aggregate modeling of selected variables render models that are easier to understand and manipulate, the goal of abstraction is not the production of a thin sketch of reality. In fact, much of the "knowledge" in a model is captured not in the detail, but in the aggregation and relationship of detail into a coherent picture. The credibility of these simpler models can be supported by direct or indirect links to more detailed models or historical data. There is, however, an increased possibility that an important factor or relationship may be omitted or lost in the aggregation process.

Ultimately, the level of detail chosen must be determined by the information required. Effective models may characterized in terms of several key features:

- Level of detail is adequate to support evaluation of principal factors of interest at the current stage of development;
- Representation scheme and mode are appropriate to the question at hand;
- Assumptions regarding the nature and relationship of the variables can be supported by valid sources (historical data, acknowledged experts, output from other validated models); and
- Model is understandable to the responsible analysts and the critical reviewers.

HCI designers use models to conceptualize the user, the tasks, and the system supports. To effectively incorporate cognitive systems engineering into HCI design processes requires a framework for integrating and extending the multiple models that support understanding, representing, and translating the decision-maker's cognitive tasks into information presentation and interaction designs.

1.2 Understanding: Decision-making in Complex, Dynamic Domains

Woods (1988) models the environmental factors that affect decision-making difficulty in terms of the levels of complexity, dynamism, uncertainty and risk inherent in the domain. This model provides a means for assessing the dimensions of a problem solving environment with respect to both the degree and the nature (sources) of difficulty. Decision-making in complex, highly-dynamic environments requires rapid comprehension of evolving situations through the processing of a high volume of information. Furthermore, having "more data" is not, in itself, an assurance of success. The effectiveness of the information use depends upon correctly matching information processing capabilities with cognitively-demanding decision-making tasks.

There are several models for human decision-making in complex, dynamic environments. For example, Rasmussen's (1986) skills-rules-knowledge (SRK) model outlines the changes in information processing and potential errors in decision-making based upon the user's expertise in decision-making tasks and the contingent nature of the current task(s). Recognition of a well-known pattern of activity might invoke a rule-based response, while novel situations would require formal reasoning based upon deeper knowledge of the underlying factors in the task and domain. Errors occur when the response level is not correctly matched to the situation – such as when a decision-maker incorrectly force-fits a rule-based solution on a novel problem. The SRK model of diagnostic processes further points out the ability of experienced decision-makers to make inferential leaps from problem diagnoses directly to correct responses without performing the sequential option generation and evaluation steps dictated by rational decision-making models.

The Rasmussen model of diagnostic problem solving is similar to Klein's (1993b) Naturalistic Decision-making (NDM) model of expert decision-making in time-stressed, high-threat situations. As in Rasmussen's model, experts recognize similarities in situations to events previously experienced. This recognition triggers candidate courses of action (COAs) for mental simulation to assess their suitability in the current situation. The central premise of recognitional decision-making in Klein's NDM model, and earlier recognition-primed decision-making (RPD) model, has been borne out in several empirical studies. For example, studies of the cognitive tasks associated with decision-making in the AEGIS Combat Information Center (CIC) indicate that as much as 87% of the decision activities involve some aspect of situation assessment (Kaempf *et al*, 1992). This emphasis is due, in part, to the existence of clear operational procedures for selecting and implementing a COA once the situation is clearly understood.

Figure 1.2 models the basic constructs of Klein's NDM/RPD model with indications of points at which cognitive errors can interfere with decision performance. First, the detection of a problem and attention to the relevant problem details is affected by limits in human attentional resources that lead decision-makers to selectively focus attention. The decision-maker's interpretation of the cues is also subject to errors in interpretation, such as preference for information that confirms the current interpretation (confirmation bias) or undue weighting of information based upon the order of presentation (primacy and recency biases) (Reason, 1990). The interpretation of the situation triggers recognition based upon a match of the key situational features to previously experienced or learned situations. This matching process is hindered by errors in long-term memory storage and analogical reasoning.

Finally, identification and selection of a course of action involves mentally simulating the possible outcome of the analogized option. In complex decision domains, mentally simulating the causal chain of decision consequences can

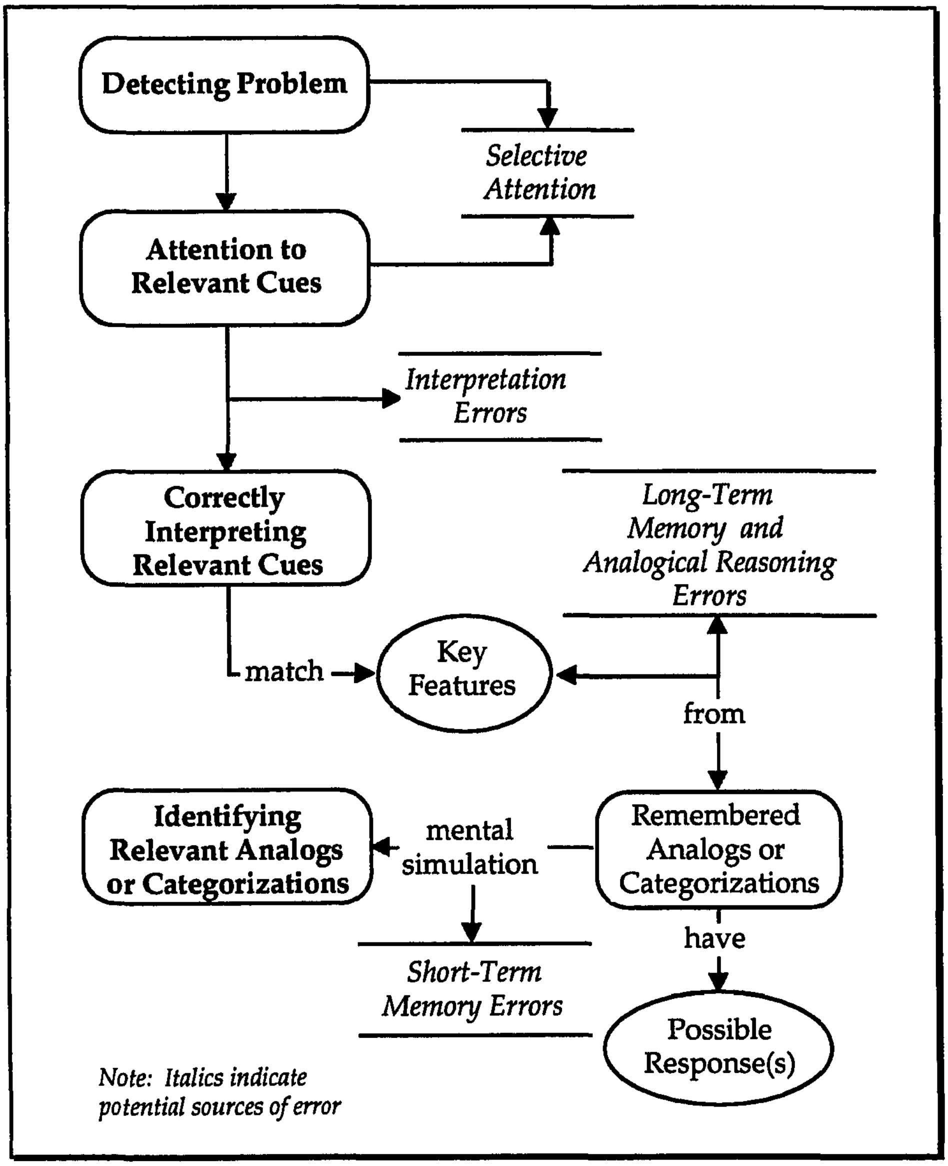


Figure 1.2: Simple Model of Recognitional Decision Processes Indicating Potential Sources for Error

exceed human short-term memory capacity. Coping strategies to minimize the increased cognitive load may result in a failure to anticipate unacceptable side-effects. These error forms are consistent with the sources of human error discussed in connection with Rasmussen's SRK model of cognitive control (Rasmussen, 1986; Reason, 1990).

Cognitive systems engineering of HCI designs for human-computer cooperative decision-making requires the ability to model the processes by which human decision-makers perceive, interpret and respond to decision situations. In addition, the HCI designer must be able to identify the possible sources of error in decision tasks and their potential effects on decision performance.

1.3 Representation: Identifying & Communicating System Requirements

The importance of improving situation assessment is echoed in the problem solving involved in system development. The early phases of software development are characterized by the greatest degree of uncertainty. As a result, as much as 80% of the mismatch between what the user wanted and what the developers delivered has been traced to shortfall in the definition of requirements (Boar, 1984). Barry Boehm's (1981) research indicates the cost to fix these discrepancies may range as high as 100 times the cost had correct requirements been identified during the requirements analysis phase. Furthermore, empirical evidence from a number of studies reveals dramatic increases in error correction costs the later in development cycle the error is found (Daly, 1977; Boehm, 1976; Fagan, 1974). The requisite rework leads to cost overruns and schedule slippage. Conversely, approaches to software development that eliminate rework and post-development modification promise productivity improvements from 30 to 50% (Boehm, 1987). For this reason, the search for cost-effective system performance improvement methods should be directed at increasing the accuracy, completeness, and precision of requirements identification and representation.

Various studies demonstrate the power of informal reviews and formal inspection to reveal as much as 65% of the errors in requirements (Basili & Weiss, 1981; Brugerre, 1979). In cooperative human-computer decision-making, the human "component" assumes responsibility for certain functions, often through a dynamic allocation that is triggered by the situation. The human tasks and computer support necessary to accomplish these functions are rarely stated as explicit system requirements, thus, are not readily available for review or inspection.

Representing the human decision-maker as a part of the functional requirements of the system has several advantages:

- *Accuracy* - incorrect assumptions, omissions, inconsistencies and ambiguities can be brought out through inspection & review for earlier detection of errors;
- *Completeness* - explicitly stated goals are more likely to be a part of the design solution; and
- *Precision* - supports the early identification of meaningful measures of performance (MOPs) and effectiveness (MOEs) for evaluation at each development phase.

While the advantages seem obvious, the real viability of this concept lies in the ability to identify and represent the human decision-maker's requirements as part of the system development process such that development time and cost are ***reduced*** through the elimination or reduction of rework.

HCI design requirements typically stop at the interface between the user and the computer system. User tasks are conceptualized and represented primarily in terms of system operation tasks or interaction task requirements (ITRs). Cognitive systems engineering requires models for defining and representing in the system requirements specification (SRS) the cognitive task requirements (CTRs) involved in human-computer cooperative decision-making as well as the ITRs. In accordance with Davis' (1993) definition of the well-written SRS, CTRs should describe what the system must do to support the decision-maker's problem solving tasks without stating how it will use hardware and software to do it. Currently, there are no models for specifying CTRs in requirements documents.

1.4 Translation: Developing HCI Concepts

Advances in information presentation and interaction technologies have provided the HCI designer with an impressive range of tools and techniques to design and engineer systems for supporting human-computer decision-making. Research findings in the cognitive sciences and information technology continue to inspire design guidelines for improved application of technology to decision support. Furthermore, this research bears out the increased possibility of error in human performance when fundamental principles are ignored in HCI design. For example, poorly designed interface/interaction approaches have been identified as resulting in degraded problem solving performance due to

- a decrease in the field of attention, or cognitive 'tunnel vision' (Bainbridge, 1987);
- failure to process important information (Klinger *et al,* 1993; Woods, 1984);
- getting lost in the network of displays, menus, and windows (Elm & Woods, 1985); or
- an increase in mental workload (Goldsmith & Schvaneveldt, 1984).

Cognitive systems engineering incorporates experimental and empirical research findings to design human-computer interaction to aid the cognitive processes involved in decision-making and problem solving. Keren (1990) describes two types of cognitive aids: procedural and structure modifying. As presented in Figure 1.3, these two approaches represent the end points along an aiding continuum that trades off the range of applicability against the cost and effort to develop the aid. Procedural aids provide specific direction in the form of a prescribed procedure or training with feedback to correcting a narrowly defined cognitive error.

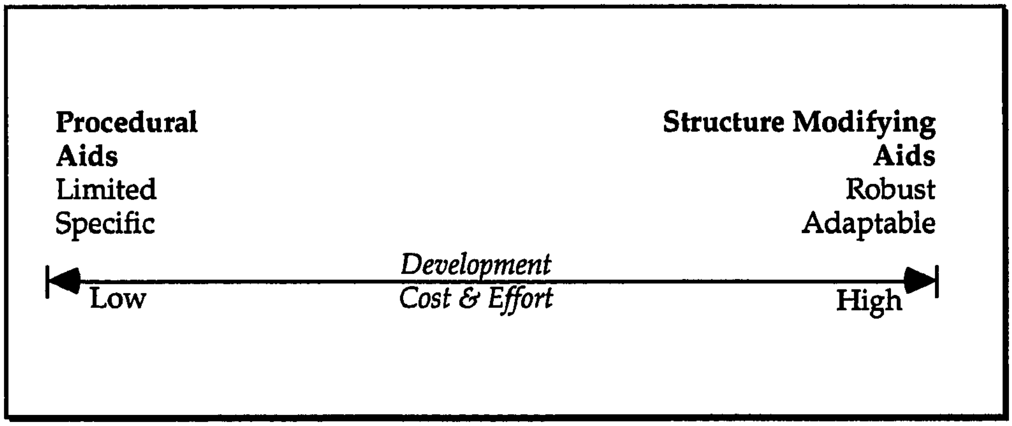


Figure 1.3: Cognitive Aiding Continuum

Procedural approaches constitute the majority of de-biasing methods described in Fischhoff's (1982) critical survey. The chief advantage in the procedural aid lies in its low-cost efficiency within a well-defined scope. Applying the procedural aid does not require, or necessarily promote, acquiring a deeper understanding of the task or cognitive error. In this sense, the procedural aid acts as a prosthesis designed to alleviate the effects, or symptoms, of the cognitive error without necessarily treating the underlying cause of the condition. The narrow focus of procedural aids limits their application to specific tasks under specific situational contexts. In more complex, dynamic decision environments, the procedures may prove brittle with a resulting potential to degrade decision performance. For example, a procedural method (i.e., training and feedback) employed during practice trials to reduce the overconfidence bias in a specific task tended to induce an anchoring and adjustment bias that persisted in subsequent experimental sessions (Arkes *et al*, 1987).

At the opposite end of the continuum, structure modifying aids, or "restructuring" methods seek to guide the decision-maker's cognitive representation of the problem by making knowledge explicit (Fischhoff, 1982). Such methods involve a qualitative change in the decision-maker's problem understanding and structuring to effect performance improvement. For example, Koriat *et al* (1980) linked decision-maker overconfidence with a tendency to seek evidence confirming the preferred choice while disregarding or devaluing contradictory evidence. They successfully mediated this tendency by requiring decision-makers to itemize the evidence ***for*** and ***against*** the chosen option prior to assigning a confidence level to their judgment. The resulting confidence levels were more consistent with the predictions of a normative certainty model. These findings were supported by a similar studies, such as Hoch (1985).

The deeper knowledge inherent in the more broadly applicable structure modifying aids typically entails higher development cost and effort. This results in the classic tradeoffs weighing cost and schedule demands against functionality and performance goals. Given that the investment lies primarily in the requirements and design phases of development, providing more cost-effective means for incorporating the requisite knowledge acquisition and design translation techniques into the development process should reduce the imbalance in those tradeoffs.

Cognitive systems engineering requires models for translating decision-makers' cognitive task requirements into information presentation and interaction designs. Theories of cognitive, or mental, models provide one means of conceptualizing the relationships between how decision-makers think about cognitive tasks and the corresponding HCI design concepts that support those tasks. Norman (1983) defines the means by which designers and users understand and interact with computer-based systems in terms of the construction and use of multiple mental models. The concept of a "mental model" appears with various definitions, taxonomic structures and applications in the cognitive science literature. Carroll and Olson (1988) review the mental model literature and offer a practical definition of mental models. In their definition, a mental model

- incorporates "a rich and elaborate structure;"
- involves an "understanding of what the system contains, how it works, and why it works that way;" and
- provides a way "try out actions mentally before choosing one to execute." (Carroll & Olson, 1988, p. 51)

The cognitive science literature presents numerous descriptive theories and empirical studies that attest to the existence of mental models (c.f. review in Staggers & Norcio, 1993); however, there remains no systematic method for satisfactorily harnessing the power of mental models to guide the design of HCI for decision support. Green (1990) suggests that the current practice of HCI research and design built upon numerous unlinked generalizations should be replaced with a requirements structure linking a set of limited generalizations and theories. A "limited" theory or generalization is a narrowly defined construct specifically applicable within the context of the requirements. For example, using a limited theory of mental models to explore a user's understanding of how to accomplish a task provides a practical framework for applied research in HCI designs to support that task without attempting to address the larger questions posed by basic research. Rasmussen's (1990, 1986) skills-rules-knowledge (SRK) model of cognitive control is an example of the application of a framework of limited theories of mental representation.

In complex, dynamic environments, the interaction models required for human-computer cooperative decision-making must assist the decision-maker in maintaining situational awareness and understanding the short- and long-term consequences of decisions. This implies a framework of models in the mind of the user that must be represented in the interaction and interface design. These include:

- **task interaction models** - representation of the current state of the target domain (situational awareness), means for acting on the domain (task variables), and means for predicting the consequences of actions on the domain (outcome simulation); and
- **system interaction models** - representation of the current state of the system and the means to understand the actions required to perform tasks using the system.

Carroll *et al* (1988) propose a structured methodology for designing effective interface metaphors that provides a useful starting point for developing interaction models. Extending this method to the design of HCI for decision aiding suggests the following basic activities:

- Identify potential task domain models - e.g., network models for route planning;
- Describe the match between models and the domain in terms of user task scenarios - i.e., the constraints and affordances implied by the analogy;
- Identify the potential mismatches and their implications - i.e., where are the gaps or breakdowns in the analogy; and
- Determine the appropriate design strategies to help users manage unavoidable mismatches.

The human decision-maker's access to pertinent decision information is a key aspect in supporting rapid, accurate situation assessment and decision-making. Representational aids allow the user to identify the relevant information in a complex dynamic environment, visualize the semantics of the domain, and manipulate critical decision variables (Woods, 1991). For example, Bennett *et al* (1993) demonstrated performance improvement in complex, dynamic decision tasks through the use of configural displays. Configural, or object, displays represent the high-level, or semantic, relationships of the domain in terms of individual low-level elements. In related research, Vicente and Rasmussen (1992) propose system designs in which the key decision variables and their relationships to the decision tasks are integrated in what they term an ecological interface design (EID). They cite several case studies that demonstrate the improvement of control system diagnostic tasks through the introduction of ecological interfaces. Similarly, MacMillan and Entin (1991) demonstrated improved decision performance on a missile threat evaluation and launch decision when expert decision-makers were provided "decision-oriented" displays rather than "data-oriented" displays. It is important to note that each of these information display approaches has also been shown to compromise performance when the decision requirements were incorrectly or inadequately identified and translated to display designs.

A substantial body of literature exists and is readily available to support the translation of individual information presentation and interaction requirements *once they are identified and understood* (e.g., DOD, 1992; Gardiner & Christie, 1989; Smith & Mosier, 1986; US Army Research Institute, 1989). These guidelines are not intended to specify designs, rather they serve to move the design process from a "black art" to a craft which may be examined and understood. In addition to text-based design support, the iterative construction and evaluation of HCI design prototypes helps to refine requirements and evaluate designs. Prototyping is a widely accepted software systems engineering method for capturing evolving requirements in complex systems (Davis, 1993; Wilson & Rosenberg, 1988). Computer-based support in the form of computer-aided software engineering (CASE) tools and pre-defined interaction structures permit the rapid construction of prototypes for communication with other members of the development team, the sponsors, and the end-users.

This research outlines a framework for employing cognitive systems engineering principles and practices to enhance the requirements identification and design phases of system development and improve human-computer interaction (HCI) designs for decision support. The framework focuses on the application of research and technology in developing a more comprehensive *understanding, representation,* and *translation* of the decision-maker's cognitive task requirements in human-computer cooperative decision-making. Additional guidance is provided in a series of tables (Appendix B) summarizing research from software engineering, decision sciences, cognitive psychology and other related fields. These supports assist the designer in defining a more robust set system requirements and guide design tradeoff decisions.

Following the presentation of the CSE framework, a system design case study in human-computer cooperative decision-making demonstrates the practical implementation of CSE for HCI design. The case study leads the reader through the application of the guidance tables to a "real world" design problem. The case study also presents an empirical evaluation of the benefits of the design framework in terms of

- improving the ***process*** of creating information presentation and interaction designs and, consequently,
- improving the HCI design ***product*** to increase the functionality of the delivered system and enhance decision-making performance.

Finally, the prototype for a CSE design practitioner's handbook (Appendix A) is presented to suggest a format for making the CSE design framework usable in the information systems engineering community.

2. A Cognitive Systems Engineering Framework for Human-Computer Interaction Design

2.1 Overview

The identification and analysis of decision aiding requirements and the design of human-computer cooperative systems to address those requirements is a process of creating and refining models. The models involved encompass various aspects of the problem domain and evolving technological solutions. Requirements documents are text-based models of the operational need; software and hardware designs are text and graphic models of the solution path proposed. Prototypes are also models, representing the current design of the system being developed. In between are many more models created in data structures, drawings, charts, etc. Structuring, evaluating and refining these models highlights gaps in the requirements or design and alerts the designer/ developer to the critical factors for successful performance. The CSE framework for HCI design is a guide to creating, structuring, and applying a series of models to accomplish the development of human-computer interacion design for a system to support human decision-making.

Systems and software engineering literature abounds with various life cycle models. Central to most of the accepted models is the notion of a structured, iterative process beginning with the identification of a problem and ending with the delivery of an operational system. Although development is modeled with discrete phases and flows, it is generally understood that the actual processes overlap and some may occur in parallel. The various models differ in terminology, phase boundaries, and the level of detail presented; however, most identify several common life cycle stages:

- problem definition
- requirements identification
- design
- implementation
- testing and evaluation
- operational fielding and maintenance

These six phases are approximated in Boar's (1984) structured development life cycle (Figure 2.1) and Andriole's (1987) systems design methodology (Figure 2.2). With only cursory examination, these models would seem to confine evalu-

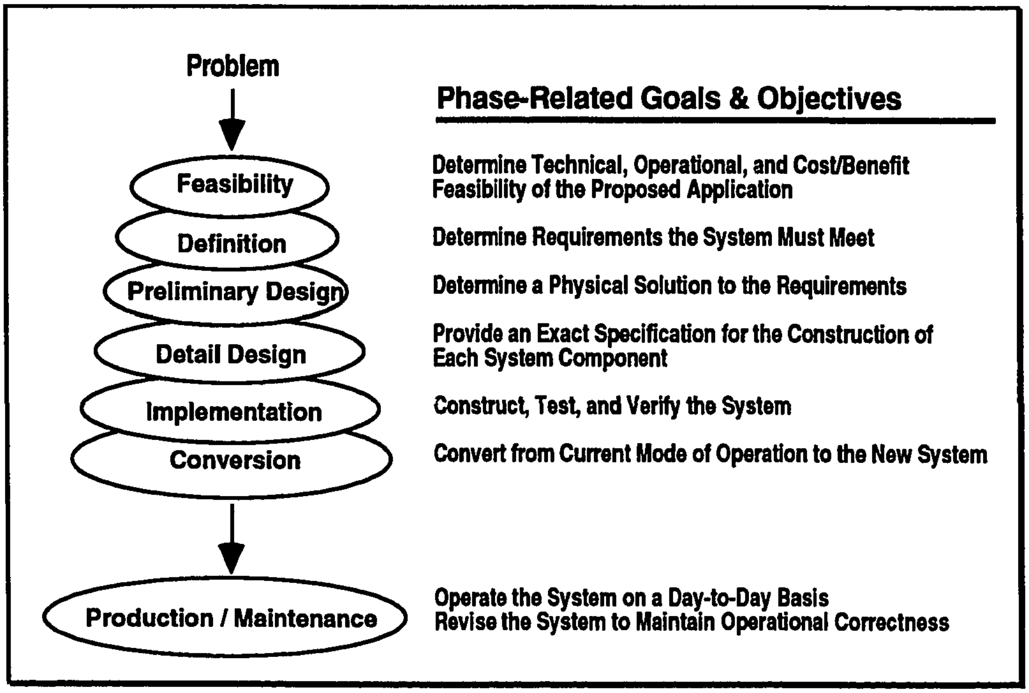


Figure 2.1: Boar's Structured Development Life Cycle
(adapted from Boar, 1984)

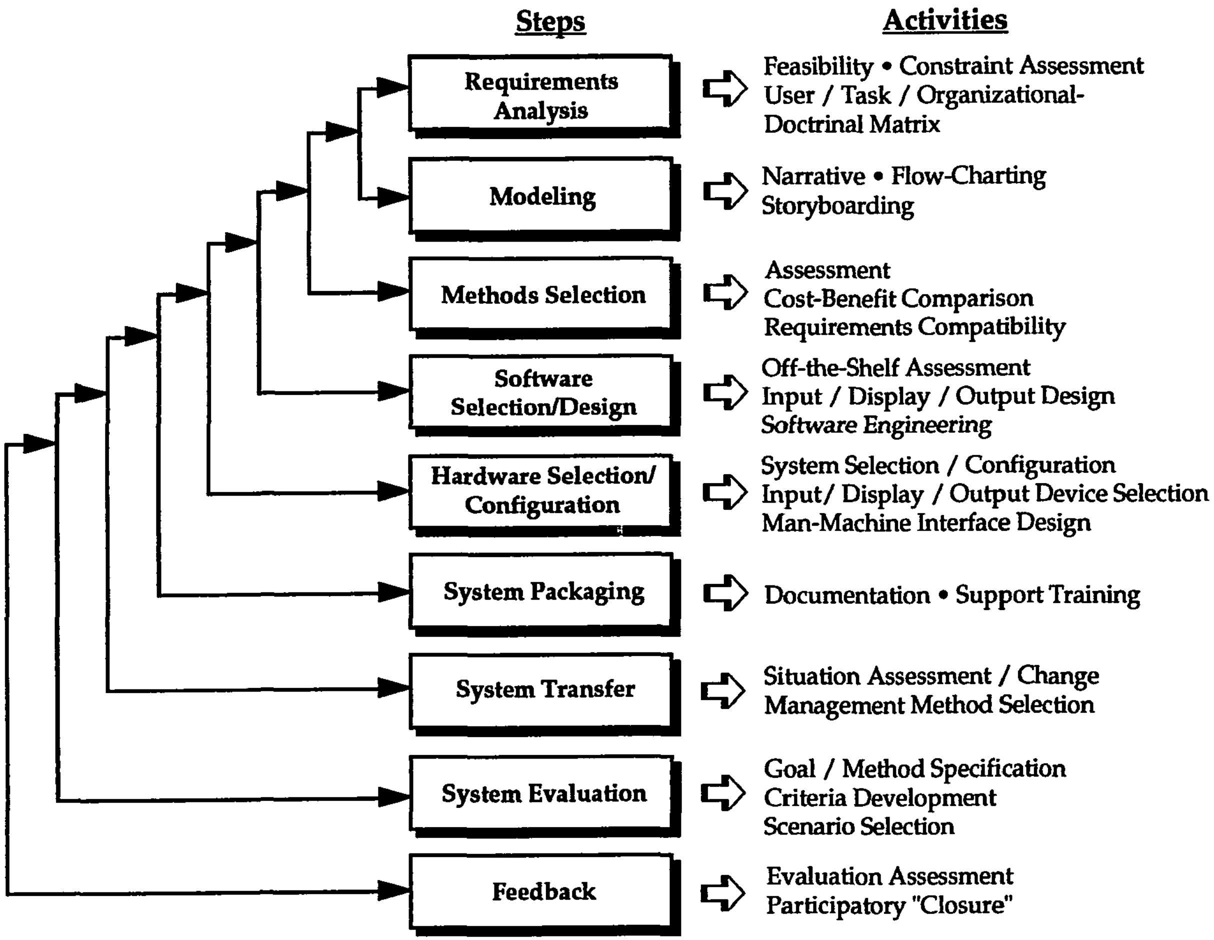


Figure 2.2 Andriole's Systems Design Methodology (Andriole, 1987)

ation activities primarily to the last stages of development. Adelman (1992) points out, however, that judgments and decisions pervade every phase of the development process. The results of analysis and evaluation (represented as feedback loops in Andriole's model) provide input to support development objectives at each stage and to determine whether those goals have been achieved. This continuous evaluation is a critical component in requirements-driven design.

In evolutionary design and development processes, prototyping has become an important tool for identifying user requirements and providing feedback on the working design against the requirements (c.f., Arthur, 1992; Connell & Shafer, 1989). To assist the development of decision-oriented displays, Metersky (1993) proposes an iterative prototyping approach to system design and development that highlights the requirements of the human decision maker. Andriole (1990) presents the requirements and design prototyping process as a miniature version of the larger system development process. In similar fashion, the CSE framework proposes an iterative sequence of activities to support the development of HCI design prototypes that correlate with traditional systems development phases (Figure 2.3). The goal of the framework is to coordinate the HCI design with the other development activities and better utilize the information resources collected during each phase to enhance the design of the system. This connection promotes smoother integration of prototyping activities and findings into the overall development effort. The information inputs, sub-tasks and process outputs for each of the six phases are explained below.

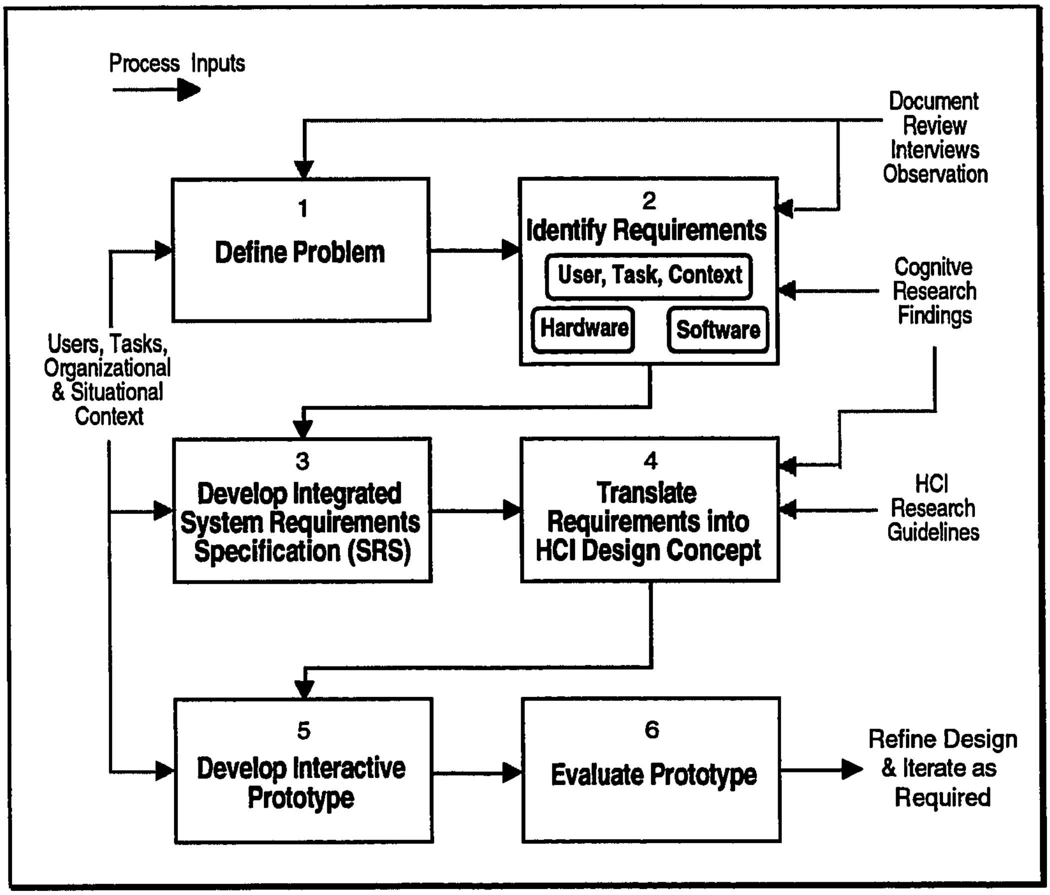


Figure 2.3: The CSE Framework for HCI Prototype Design and Development

2.2 Phase One: Defining the Problem

2.2.1 Goals of the Problem Definition Phase

The problem definition phase serves two purposes. First, the definition phase determines the scope of the proposed system in terms of what is needed and technically feasible. Second, this initial phase establishes the goals and objectives for the system development effort. As illustrated in Figure 2.4, problem definition is accomplished by examining three general types of information:

- *System Context* - who will use the system, what they are trying to do with it, under what conditions it will be used, etc.
- *Constraints* - "built in" requirements for inputs, outputs, interconnection, environmental tolerances, etc.
- *Technological Opportunities* - leverage points where technology may be applied with greatest benefit.

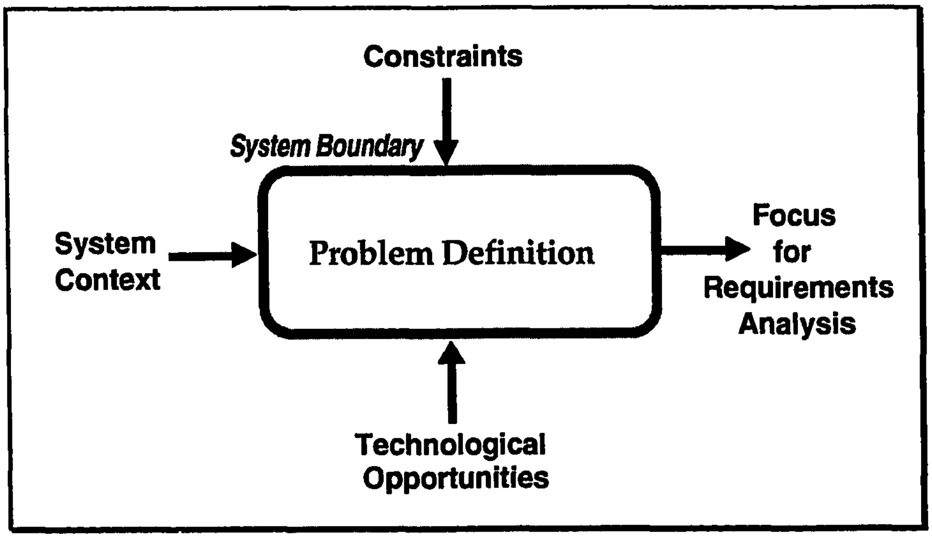


Figure 2.4: Issues in the Problem Definition Process

In systems involving human-computer cooperative decision-making, each of these aspects of the definition has important implications for the HCI design.

During the initial definition phase, the design team gathers information to understand the functional goals of the system as defined by the sponsoring organization. Information drawn from various organizational documents and discussions with the sponsor help to sketch the system boundaries and develop a profile of the system context as defined by:

- **users** - experience, training, organizational roles;
- **tasks** - high-level functions, performance goals, decision task characteristics (timing, criticality);
- **organizational context** - organizational goals, missions, control structures, communication modes; and
- **environmental context** - when, where, how, and under what conditions will the system be used.

This information comprises the operational need the new system must meet. The various dimensions of the system context each generate constraints on the system that must be explored during the requirements phase and addressed in the design. Moreover, constraints involving human performance, hardware, and software interact. For this reason, it is essential that the HCI design team coordinate with the other members of the design team during this early definition to consider these interdependencies.

Both new system development and system re-engineering provide opportunities for applying new technologies. Many of the significant advances in HCI technologies, particularly those involved in information presentation, have become more feasible for operational systems due to innovations in chip architecture. As a result, HCI design concepts that were once only laboratory show pieces are now cost-effective for use in fielded systems. Initial decisions regarding the system concept trade off these technological opportunities (i.e., what ***might*** be done) against the system context and constraints (i.e., what ***must*** be done). The HCI aspects of the definition both affect and are affected by the other hardware and software issues.

2.2.2 Models for Problem Definition

The problem definition phase provides the initial signposts to guide the more detailed requirements identification and analysis that follows. For this purpose, the most useful outputs from the definition phase are preliminary models, such as concept maps and functional decomposition diagrams, defining the central constructs of the system and indicating relationships between them. One of the most difficult aspects of the definition process is the internal (and sometimes external) pressure to "define" in terms of solutions. Jumping to solution thinking during this phase may focus the subsequent requirements identification activities on a subset of the problem while neglecting other equally relevant aspects. This "tunnel vision" early in the development can lead to one of the most common sources of error -- defining the wrong problem and then proceeding to solve it.

Problem definition and requirements identification activities vary widely in the granularity of representation required. The same design may use different modeling methods for different development efforts. Some models are suitable for extension and elaboration as the design evolves, while others are more narrowly focused with limited application. Several methods specifically address the semantic aspects of domain knowledge and are useful to the HCI designer. For example, concept mapping (Gowan & Novak, 1984) is an informal technique for modeling relationships and interdependencies. The method was developed in the field of educational psychology and has been applied successfully the acquisition and modeling of knowledge requirements for decision support systems (Klein, 1993; McFarren, 1987). Kieras (1987) developed a similar set of goal-task models to structure cognitive learning tasks. This method was used to identify and structure the cognitive requirements for embedded training in tactical information systems (Williams, 1989). Cognitive mapping (Montazemi & Conrath, 1986) is a more formal technique that evolved in the field of artificial intelligence. It focuses on modeling cause and effect relationships for process or behavior understanding and has been adapted to create computable cognitive architectures in neural networks (c.f., Zhang *et al*, 1992).

Byrd *et al* (1992) survey eighteen requirements analysis and knowledge acquisition techniques that facilitate problem domain understanding in terms of information requirements, process understanding, behavior understanding and problem frame understanding. They emphasize that none of the methods is suitable for eliciting and modeling all the dimensions of domain knowledge. The key to effective problem definition is finding a means for creating and relating *multiple* models, or views, of the problem. When the problem is complex and multi-dimensional, the design team needs methods specifically designed to facilitate interdisciplinary thinking. For example, multi-perspective context models, such as those described for problem analysis in Davis (1993), assist in creating informal models for review and iteration with the sponsors and operational users. Similarly, Zahniser (1993) describes the creation of N-dimensional views of the system developed by cross-functional development teams. The process is designed to encourage innovative thinking and bring multi-disciplinary experience to bear on system development problems.

Problem definition models help to organize the system goals and objectives to guide the developers in the requirements identification phase. The CSE framework does not specify or require any particular modeling method; rather it is left to the developer to ascertain which methods will best address the issues of interest. For the HCI design team, the most relevant issues are those aspects of the problem definition that address the functional roles and activities that are modeled for the human users. Using the *initial high-level function allocation*, the HCI team must begin to identify and analyze the human task requirements and the associated implications for human-computer interaction.

2.3 Phase Two: Identifying and Understanding Requirements

2.3.1 Goals of the Requirements Identification and Analysis Phase

During the requirements identification and analysis phase, the HCI design team focuses on deepening and extending the knowledge represented in the problem definition models with respect to the human users and their task support needs. The HCI design requirements provide a focal point for integrating the information gathered on the users, problem solving tasks, and the decision environment to guide design decisions involving interaction control and focus of attention. These requirements include not only the interaction task requirements (ITRs) that define the operation of the interface, but also the cognitive task requirements (CTRs) that define the supports for human decision task performance. Particularly in cases where the decision tasks are complex and must be performed in a dynamic, time-stressed environment, the operation of the interface must not distract the decision-maker from the primary tasks involved in accomplishing the organizational goals. The HCI designer uses the CTR and ITR information to determine the most beneficial information representation modes, display formatting, and information interaction routines.

During the requirements analysis phase of development, the cognitive task requirements (CTRs) of the user can be identified and defined as part of the normal requirements identification activities. The goal during this phase is to gain an understanding of the functional tasks the human user/decision-maker must perform and how those tasks are defined and affected by the user, the organization, and the situation. Using the high-level conceptual models from the early problem definition activities and the evolving hardware and software requirements, the HCI designer develops models of information flows, task allocations, and organizational procedures for decision-making. At this point, it is useful to observe the way the organization currently addresses the problem and interview representative users to expand and correct the preliminary functional, procedural, and dependency models.

The CSE framework uses the information gathered for the system requirements analysis and expands it to include a model of the user's cognitive tasks (as implied by the information flows or prescribed by operational procedures) and analyzes that model with respect to the user's information requirements and the possible sources of cognitive errors. The CTRs are constructed through the process of evolving and relating models that profile the user and organization, describe the environmental and situational context, and define the various cognitive tasks involved in accomplishing the functional tasks assigned to the human-computer decision component (Figure 2.5).

2.3.2 Models for Requirements Capture and Analysis

A CTR represents either the nature of the *input* required for a human decision making task or the content of the *output* required from that task. Thus, the initial objective of the requirements phase is to identify the kinds of cognitive tasks the users may be required to perform and examine the factors affecting performance. If a task affects decision performance, it is necessary to find out what characteristics of the task do so. Meister (1981) identifies five task dimensions that may affect performance:

- Functional requirements (cognition, perception, etc.)
- Complexity
- Mental workload

- Temporal factors (pace, duration, sequence, etc.)
- Criticality

Cognitive task taxonomies, such as those found in Fleishman and Quaintance (1984) and Rasmussen *et al* (1990) can be used as a filter to identify and categorize basic cognitive tasks with respect to these dimensions. In addition, Andriole and Adelman (1989) present a taxonomic discussion of human information processing and inferencing tasks with respect to the potential cognitive errors associated with each.

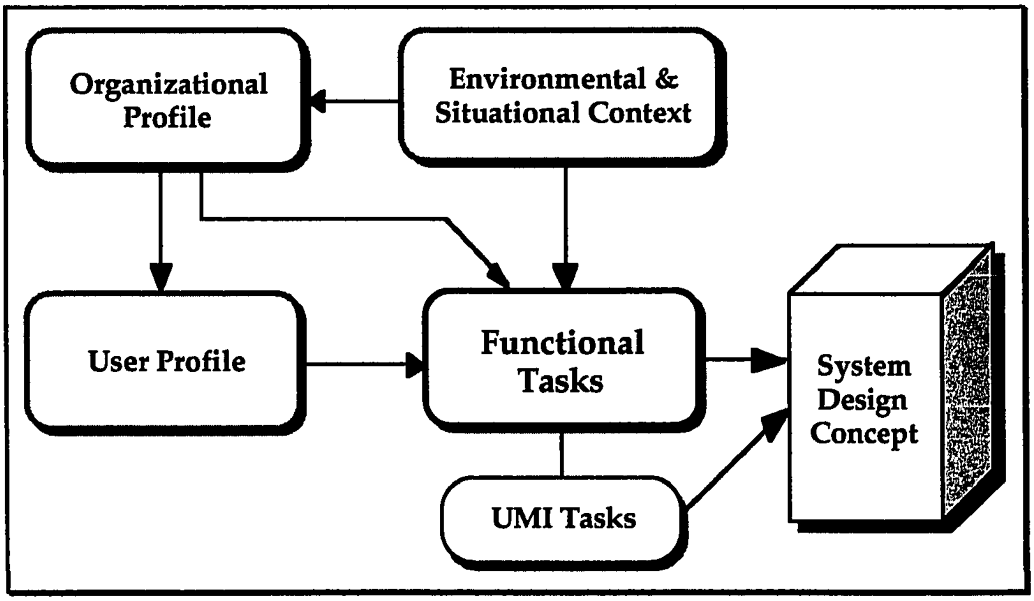


Figure 2.5: Modeling Resources for CSE Requirements Identification

As the design team reviews the context diagrams, functional decomposition diagrams, and strawman storyboards, descriptions of activities can be examined for verbal constructs that indicate human decision-maker actions. For example, in systems where the human decision-maker must *monitor* a situation and *interpret* evolving events, the software designers may view the inputs to the user as updates to a data base. From the user's perspective, however, this requirement has implications not only for interface operation design, but also for the information presentation design. In order to interpret those updates, the changes must not only be visible to the user, but also presented within a meaningful context. Using the concepts of analogical representation and causal reasoning, this context might include some mapping of relationships between key factors, tracing of changes in relevant factors over time, and/or models of a goal state to which certain parameters should conform. At this point, the information presentation and interaction requirements continue to be identified from the user's perspective without specifying the design solution.

Figure 2.6 models a simple decision task by relating the incoming information and the human information interpretation process. For the HCI designer, this model helps to identify the elements, or key variables, that need to be presented to the user (e.g., Factor F and Factor Z values). It also indicates that the user is basing part of the interpretation of this information on the potential change in Factor F values across time.

This simple model raises numerous questions for both the HCI design and the support system, such as:

- How often should the data be updated?
- How does the decision-maker need the information presented to comprehend the meaning of the change?
- Does the decision-maker ever need to know or review values of Factor F going back several updates? If so, is the current direction of the design implying that the decision-maker will retain this in his/her memory or keep notes off-line?
- Does the decision-maker make these interpretations routinely? Occasionally? Rarely?

- How does the change in Factor F relate to Factor Z?
- Will the decision-maker have experienced a wide or narrow range of interpretation situations?
- What situational contingencies might negatively affect the decision-maker's accurate interpretation of these factors?
- How does the decision impact the mission? How critical is it? How rapidly must the decision be made? Where and how will it be disseminated?

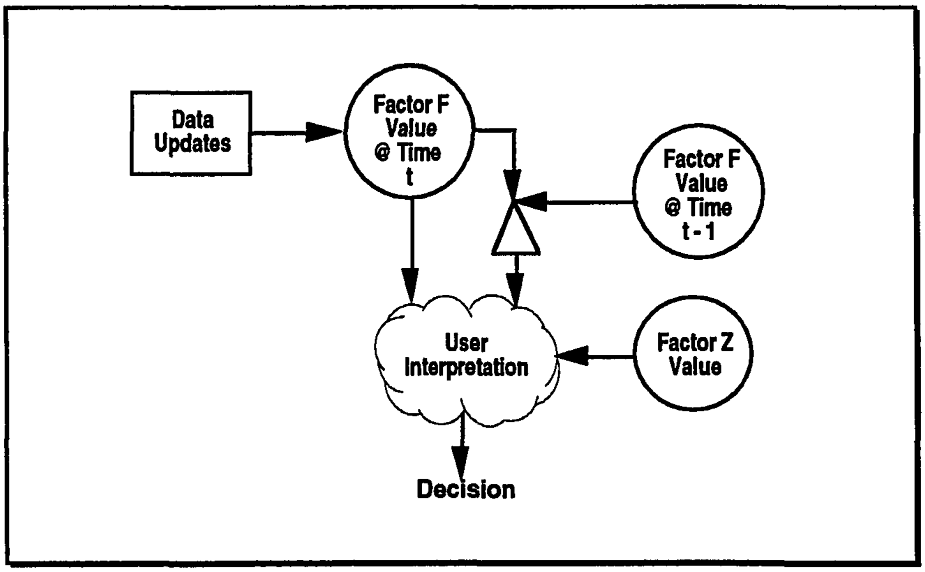


Figure 2.6: Model of the User's Information Processing and Inferencing Activities in an Example Decision-Making Task

These questions and others may need to be addressed in the design and coordinated with the other development teams. To answer them requires understanding not only the structure of information flows, but also the way in which that information is used.

The task analysis is the focal point of the HCI requirements models; however, there is no method for capturing and analyzing tasks that fully addresses the range of task factors and questions. Since the ultimate objective of this process is the application of task analysis findings to the design, development, or evaluation of the target system, this requirement should drive the selection of suitable methods. Stammers *et al* (1990) identify a range of task analysis methods defined by their representation techniques (e.g., hierarchical, network, and flow chart methods) or by their content (e.g., cognitive and knowledge description, taxonomies, and formal grammars). The advantages and disadvantages of each category are discussed below and summarized in Table 2.1.

Hierarchical Methods

Among the best documented and most widely used analysis methods are those that hierarchically decompose tasks into their component subtasks. The decomposition process facilitates modularity and allows the analyst to control the analysis focus and level of granularity. Hierarchical representations are relatively easy to learn and understand, making them particularly useful as a means of communicating concepts to decisionmakers and other members of the development team. Similarities between hierarchical task models and those used to model hardware and software requirements make it possible to extend models developed elsewhere and model links between tasks and architectural functions.

Hierarchical Task Analysis (HTA), used extensively in process description for industry, employs a process of *progressive redescription* that successively breaks down tasks and subtasks into finer detail until the stopping criteria are met. These stopping criteria are based upon risk factor computed from the *cost* associated with non-performance of the subtask multiplied by the *probability* of performance failure. The determination of both cost and probability factors is often highly subjective and requires expert judgment. HTA diagrams also fea-

Method	Advantages	Disadvantages
Hierarchical Methods	• Provides systematic & complete structure for analysis • Broadly applicable • Well-documented; easy to learn & apply	• Difficulties with representation of parallel activities • Limited representation of cognitive factors
Network Methods	• Models temporal order; can represent parallel activities • Analysis may be performed quickly given the right data • Relatively easy to learn	• Limited applicability and narrow focus • Does not consider underlying cognitive or behavioral relationships
Knowledge Description & Cognitive Methods	• Provides methods for characterizing cognitive tasks not found in other TA methods • Consistent structure for representing task information	• Difficulty assuring completeness in highly cognitive tasks • Requires considerable analyst skill in knowledge elicitation • Expert sources may not be able to adequately verbalize knowledge
Taxonomic Methods	• Explicit categorization of task information; variety of uses • Well-documented • Relatively easy to learn & apply	• Difficult to assure completeness and definition of mutually exclusive categories • Potential for inconsistent allocation of task elements
Formal Grammar Methods	• Rigorous, formal specification of procedural task information • Provides a mapping of tasks to actions for HCI dialogue	• Narrowly focused, inflexible structure • Does not model relationships between task elements
Flow Chart Methods	• Models parallel user/system tasks and information flows • Well-documented; relatively easy to learn & apply	• Difficult to assure completeness and definition of mutually exclusive categories • Potential for inconsistent allocation of task elements

Table 2.1: Task Analysis Methods Comparison
(adapted from Stammers *et al.*, 1990, Meister, 1985)

ture annotations identifying various temporal aspects of tasks. This method has been applied successfully in the areas of training and human reliability assessment.

The GOMS (Goals, Operators, Methods, and Selection) model, developed by Card, Moran and Newell (1983), is probably the best-known hierarchical model for analyzing the behavioral requirements for procedural information processing tasks such as text editing. The major constructs of the model include:

- **Goals** - objectives of a task and subtask
- **Operators** - elementary actions necessary to accomplish goals
- **Methods** - sequence of operators and subgoals used to achieve a goal
- **Selection Rules** - rules for choosing between alternative methods

The GOMS model provides a means of predicting task completion time; however, the prediction is not robust enough to address delays due to errors or interruptions. Kieras (1988) presents a GOMS-based methodology for user interface design that supports prediction of human performance, learning time estimates, and execution time estimates. Irving *et al* (1994) applied the GOMS model to the operation of the flight management computer on commercial aircraft. To the extent that tasks are procedural and the GOMS components identifiable, this method is useful for guiding HCI design and analyzing existing systems or prototypes.

Network Methods

Network methods are appropriate for the examination of such task dimensions as temporal factors, certain workload features, and spatial relationships. Network paradigms are also useful for describing communication flows, including human ↔ human, human ↔ machine, and machine ↔ machine. Several objective methods support network analysis, including a variety of time-event charting methods, FROM-TO charts, and link analysis. Network methods appear to have some utility in the design of environments to support team performance. The most often cited study is Chapanis' (1959) redesign of a battle cruiser command post to facilitate optimal communications. More recently, there has been some interest in applying Petri nets and similar representations to the modeling of command and control tasks (c.f., Perdu & Levis, 1993; Levis *et al*, 1994). Network methods, however, do not address the underlying cognitive and behavioral factors in decisionmaking tasks.

Cognitive and Knowledge Description Methods

The traditional methods for task analysis generally do not address the cognitive processes and knowledge requirements of tasks. The development of expert systems and other knowledge-based decision support systems demanded the systematic framework for elicitation and representation of the knowledge that defines both the decision domain and the decision processes. As few of the relevant task elements are overt or otherwise observable, most of these methods employ a variety of subjective techniques, particularly the detailed elicitation techniques such as the critical incident method, verbal protocol analysis and verbal probe. Methods for describing knowledge or cognitive processes are, thus, limited by their dependence upon subjective assessments and the ability of experts to verbalize their decision processes.

The concept mapping and cognitive mapping techniques discussed in the previous section are also useful for more detailed requirements identification and modeling. The conceptual models developed during the problem definition phase provide an initial framework that may be elaborated with further detail and extended to include special purpose models. For example, specific types of decisions may be explored with diagrams that model the belief structures or cause and effect relationships that influence the decision.

Cognitive Task Analysis (CTA) or cognitive work analysis uses information from verbal protocols to analyze the skilled operator tasks involved in large scale control processes (Rasmussen, 1986; Rasmussen *et al*, 1990). Rather than analyzing the operator's information processes, CTA provides a framework for organizing the sequence of 'states of knowledge' representing what operator knows about system operations at a given point. The schematic representation provides a systematic means for modeling human-machine interaction in highly automated systems. This technique appears to have great utility in HCI design and evaluation for decision aiding systems, particularly those involving both team decision-making and the allocation of a significant number of information processing functions to machines.

The MOHAWC project (Models of Human Activity in Work Context) at Risø Laboratory links cognitive work analysis models to the design of "ecological" interfaces (Rasmussen & Pejtersen, 1993). The goals of this project are similar to those for the CSE framework. Rasmussen and Pejtersen propose "maps of design territory" to aid in bridging the gaps in current HCI guidelines. These "maps" relate various work activities to domain characteristics to provide guidance for interface design. Although the method seems appropriate for modeling and analyzing the cognitive support required for complex decision tasks, the available application examples are limited to human-machine interaction tasks.

Task Analysis for Knowledge Description (TAKD) was originally designed as a means for organizing knowledge in training applications (Johnson *et al.*, 1984). This method has been used recently in usability testing for interface designs. The representation employed is similar to formal grammar methods and attempts to abstract task knowledge independent of the specific task. The goal is to increase the generalizability of the results to make them more useful for HCI design.

Taxonomic Methods

In most cases, some taxonomy of behaviors lies at the root of all task analyses. For example, the tables in Appendix A constitute a synthesis of requirements issues in a taxonomic form. A large number of general taxonomies exist, providing an excellent source of descriptors for checklists. As indicated above, Fleishman and Quaintance (1984) present the most comprehensive survey of taxonomic methods, organized in terms of the types of information recorded. The principal limitations associated with taxonomic approaches stem from semantic confusion regarding the categorization or labeling of behaviors or activities.

One commonly used taxonomy in HCI requirements analysis and design is Berliner's (1964) hierarchical classification for measuring performance in military jobs. Berliner's classification specifies four processes (i.e., perceptual, mediational, communication, and motor processes), that further break down into activities with specific behaviors. It is the specific behaviors that provide the observable and measurable entry points into the classification. Berliner's method specifies the measures (e.g., times, errors, frequencies, workload, and motion dynamics) and categorizes the instruments for collecting the measures.

Formal Grammar Methods

One of the principal appeals of formal grammars is their ready translation into machine-understandable statements. This feature is useful in the development of expert systems and other knowledge-based applications. One of the primary advantages of formal grammars is their reduction of the ambiguity associated with more subjective approaches. However, formal grammars tradeoff completeness for precision and may not capture relevant information that does not fit into the classification scheme. Task Action Grammar (TAG) enables the direct mapping of tasks to actions and models user knowledge (Payne, 1984).

As with other methods, the TAG approach begins with task decomposition. The mapping of tasks to actions is accomplished by applying a set of rewriting rules. Although quite comprehensive in its domain, TAG is limited to the investigation of command languages and user-computer dialogue.

Flow Charting Methods

In some respects, flow charting methods resemble both hierarchical and network approaches. The ability to model both parallel and sequential activities combined with the focus on information flows, decision points, and actions make flow charting methods ideal for the representation of HCI requirements. For example, the job process chart method, developed for analyzing naval command tasks, specifies a three-level hierarchy for describing communication flows (Tainsh, 1985). The top level of the hierarchy identifies the work stations and lines of communication between them. The next level describes the tasks performed at each station. Finally, the tasks and subtasks are defined in terms of their allocation to human or machine and the subsequent HCI requirements.

Since flow charting methods employ some form of task taxonomy for identifying tasks, these methods also exhibit some of the classification ambiguity noted in other methods. Perhaps more critical in the development of decision support systems, these methods assume an unchanging external environment and a uniformity of user knowledge that undermines their validity in the target environment.

Every cognitive task performed by the human-computer cooperative decision system and supported by the HCI design is impacted by the user/decision-maker, the organizational structure and goals that define their role, and the situational environment that provides the context for their decisions. During the identification and analysis phase, the HCI design team must gain sufficient knowledge about the multiple dimensions of the requirements to model their interactions and implications for system design. The activities involved in capturing and modeling the situational context and the organizational, user, and task profiles are not necessarily discrete or sequential. These analyses occur largely in parallel and often represent shifts in focus rather than separate efforts.

The remainder of this section presents some of the relevant issues identified in cognitive research regarding the users, tasks, organizations, and situational context. Each of these requirements dimensions is presented in turn with information on the parameters of interest, including

- how they may be captured and modeled;
- how they may impact decision-making; and
- how they may be interpreted in the context of the other dimensions.

To aid the practitioner, this information is summarized in the tables contained in Appendix B following the CSE Design Handbook. These tables are referenced within the relevant sections.

2.3.3 Situational/Environmental Context

Models of the situational context, or decision environment, capture and represent the conditions under which decisions are made and the effects of agents and events external to the decision-maker and the organization. The models in this section provide several perspectives for modeling situational context and interpreting the potential impacts on decision-making. Due to their considerable interaction with the decision tasks and decision-makers, similar issues are addressed with respect to the characteristics of the decision-makers, organizations, and the tasks (see Sections 2.3.4-6). The implications of the situational context models for design are discussed in Section 2.5.1.[1]

---

[1] Tables B-1 - 4 in Appendix B summarize the key concepts defining the situational context.

Context Categories and Situational Response

Meister (1991) categorizes situational contexts in terms of four possible levels of determinacy that roughly equate to the degree to which the domain is well-bounded and predictable. The situational context may be considered ***determinate*** when the given situation or initial condition has only one significantly probable outcome. This highly predictable context for decisions includes common mechanical systems, some highly institutionalized social systems, and certain control systems. ***Moderately stochastic*** situations have only a limited number of qualitatively similar outcomes with a significant probability of occurrence. In this context, prediction of outcomes remains tractable as in the case of genetic processes or system variability due to variable dimensions in the component parts. ***Severely stochastic*** situations have a large number of qualitatively similar outcomes with a significant probability of occurrence. While event outcomes in these situations remain predictable, they are computationally intensive and beyond the range of unaided human computation. Severely stochastic situations involving human agents also have qualitative aspects that increase the difficulty of response and outcome prediction. ***Indeterminate*** situations provide so little information about possible outcomes that no outcome can be identified as significantly more probable. Meister cites psychotic human behavior and some political alliances as examples of indeterminate contexts.

These "environments" rarely exist in discrete form and decision-makers perform tasks simultaneously across a range of environments. For example, flying an aircraft requires interacting with multiple environments. The aircraft systems perform in determinate to moderately stochastic ranges. Air speed and altitude are absolute values with narrowly defined meanings for certain tasks. Other parameters (i.e., fuel consumption) represent calculated values for which there are ranges of accuracy. Outside the cockpit, the aircraft pilot must interact with severely stochastic weather conditions that may affect the aircraft in unpredictable ways. When the aircraft involved is a military aircraft, the pilot must also respond to the indeterminate environment of the battlefield.

Structure, Boundedness and Complexity in Decision Context and Tasks

In addition to the determinacy of the situational context, it is useful to understand and model the degree of structure, the boundedness and the complexity inherent in the situational context and typical decision tasks. Several researchers discuss the interaction of these factors (c.f., Fleishman & Quaintance, 1984; Meister, 1991; Rasmussen *et al*, in press) and their implications for aiding the decision-maker. The structural characteristics of the decision context and tasks should be considered in the selection of the analytical methods that form the basis of the decision aid design as well as the interaction routines that facilitate the human-computer cooperation.

Structure

The degree of structure in a decision domain characterizes the typical situations and decision tasks in terms of the extent to which information on the key variables is available and quantifiable. For example, ***highly structured*** contexts are those where all critical information is readily available and quantifiable for accurate manipulation. In ***semi-structured*** contexts, the key variables may be quantified without losing critical information or making difficult assumptions; however, often some of the critical information is unavailable. In this case, the uncertainty surrounding the decision involves "known unknowns" that may have to be inferred if further information cannot be obtained. Finally, ***unstructured*** contexts involve qualitative variables that may not be legitimately quantified. In addition, there may be "unknown unknowns," that is, critical information that is either not available or not represented in the user's model of the situation or task.

Boundedness

Closely related to determinacy and structure, "boundedness" incorporates the degree to which the key variables constrain the problem to make it tractable. The representativeness and reliability of the variables also contribute to the boundedness of the problem domain. A ***closed*** domain may be constrained and described accurately with variables that require minimal cognitive demands to manipulate. When the domain is ***semi-bounded***, the variables may only be generally representative and reliable. The associated uncertainty is manageable only by highly trained and motivated experts. The ***open***, or unbounded, context involves variables that may not be well-understood and/or reliable. The resulting uncertainty exceeds human ability to absorb and manipulate.

Complexity

The degree of complexity characteristic in the domain is interwoven in the concepts of both structure and boundedness. Woods (1988b) defines complexity in a domain or a system in terms of the number interconnecting parts or subsystems and the degree of interdependence between them. Using a structural model of situational context, complexity may be further delineated with respect to the number of hierarchical levels (vertical complexity) and number of parts or subsystems per level (horizontal complexity). In ***simple*** domains, both the vertical and horizontal complexity are low and the critical variables in the situation do not interact. In a system context, this absence of interdependence results in component functioning unaffected by performance of other system parts. In ***moderately complex*** domains, the degree of vertical and horizontal complexity increases and there is greater interdependence between the variables involved. In moderately complex domains, performance of functions may be enhanced or degraded by the performance or non-performance of other subsystems. ***Complex*** domains and systems involve many hierarchical levels extended by many interdependent parts and subsystems. The functions of a complex system cannot be performed if other subsystems perform poorly or not at all. The inherent complexity of the situational context plays a significant role in the decision-maker's ability to mentally simulate the consequences of a proposed response. From a design perspective, simplifying domain complexity may eliminate critical information with unpredictable results.

Effects of Situational Context on Decision-Making

Situational context figures prominently in several models of human information processing and decision-making. For example, Rasmussen's (1986) Skills-Rules-Knowledge (SRK) model has three levels of cognitive control based upon situational contingencies and user knowledge (Figure 2.7). ***Skill-based*** control comprises the highly integrated, automatic sensory-motor responses that occur with little conscious effort. Efficient control in this mode is dependent upon experience and a predictable environment. In ***rule-based*** responses, the decision-maker is consciously aware of taking a sequence of steps to attain a goal that may not be explicitly formulated. As a result, the decision-maker can accurately describe the procedure or rule triggered by the situation, but often cannot explain the situational cues that triggered the rule. In novel situations or unfamiliar environments, the decision-maker does not have readily understandable cues to trigger procedural responses and must use additional cognitive resources to analyze the situation. Situation assessment in ***knowledge-based***, or model-based, control is used to formulate an explicit goal and identify procedures to attain the goal. When reasoning identifies an appropriate rule or procedure, control drops back to the rule-based level. The decision-making effectiveness in this mode depends upon the quality of the decision-maker's "mental model" of the situational context.

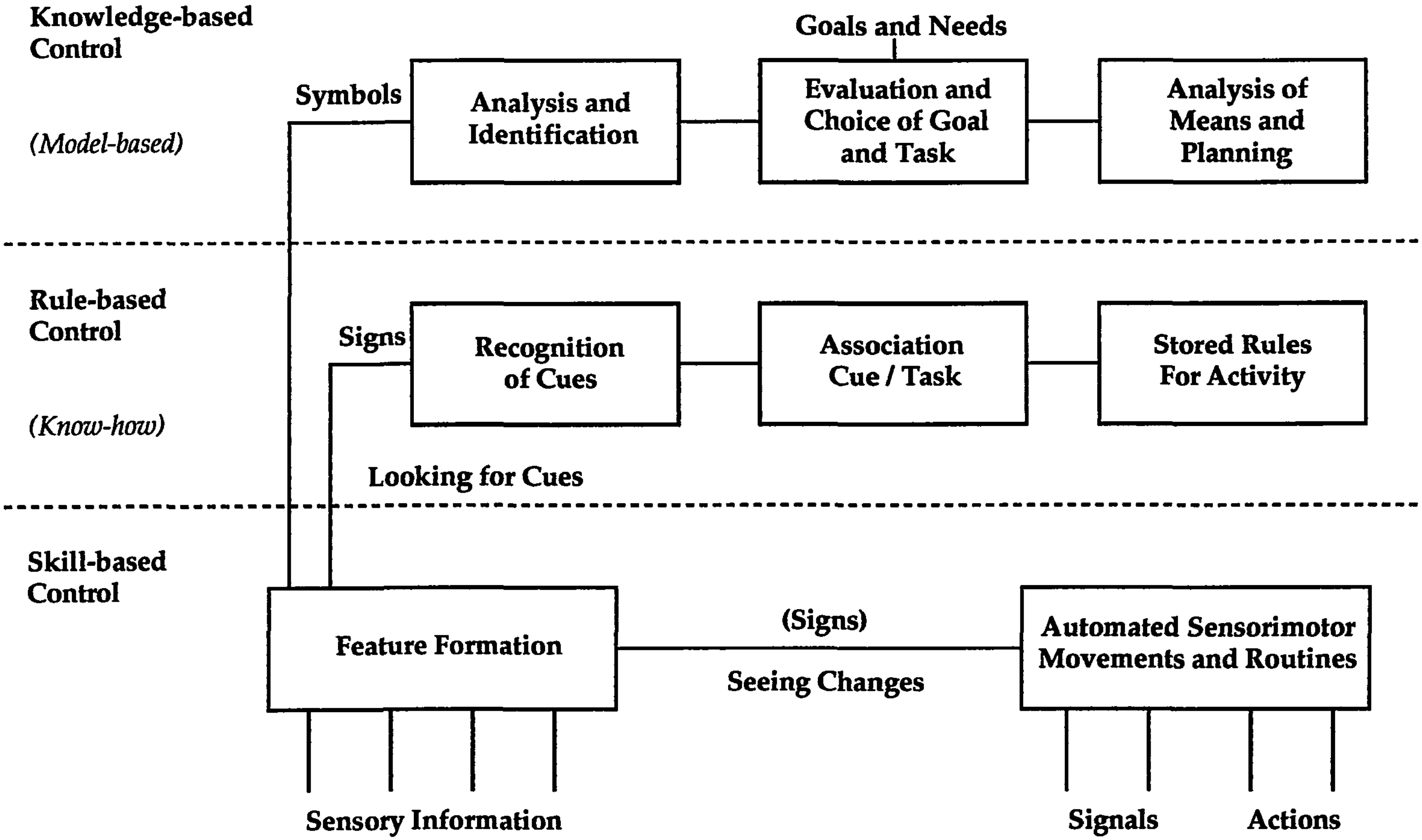


Figure 2.7: Levels of Cognitive Control in Situational Response (Rasmussen, 1986)

Understanding the situational context can provide insight into the potential cognitive demand placed on the human decision-maker. For example, in simple, primarily determinate contexts, the decision-making efforts focus on optimizing the outcome by manipulating the initial conditions. This generally involves skill-based actions and some rule-based control. Semi-structured, moderately stochastic contexts tend to induce attempts to manipulate initial conditions using primarily rule-based control. Since the possible outcomes are bounded, efforts often focus on optimizing the expected value of outcomes. In severely stochastic contexts, outcomes cannot be controlled precisely by manipulating initial conditions. Furthermore, detailed planning and reliance on pre-planned procedures (rules) are less useful due to the unpredictability of complex evolving situations. In this case, a combination of knowledge- and rule-based response control efforts focus on preparation for unfavorable outcomes and maintaining an ability to recognize and rapidly exploit opportunities. Decision responses in a complex, indeterminate situational context rely primarily on knowledge-based control. Effective performance depends upon knowing enough about the situation and the domain to classify it. The highly unpredictable nature of these contexts requires an intuitive approach based upon well-developed mental models of the domain and environment to protect against disastrous response errors.

Situation assessment and mental models also drive Klein's (1993a) Recognition-Primed Decision (RPD) model of expert decision-making in dynamic situations. The RPD model describes decision-making behaviors comparable to the rule-based and knowledge-based behaviors described by the SRK model. When forced to respond quickly in an unfamiliar situation, the expert decision-maker attempts to identify aspects of the situation similar to previously experienced situations. In simple recognition situations, matching the current situation to a previously experienced analog automatically indicates the appropriate course of action (i.e., the procedure to follow). In more complex recognition situations where there is no readily available analog addressing the key features of the situation, decision-makers must also reason about possible courses of action. This reasoning involves mental simulation of the possible outcome(s) of a particular course of action based upon the decision-maker's mental model of the situational context and ability to manipulate the network of interdependencies. The resulting cognitive demands lead to a satisficing, rather than optimizing, strategy in which the decision-maker selects the first course of action that appears to satisfactorily attain the goal.

Crises form a special case in situational contexts that impact the users, organization, and the decision tasks. Hermann (1972) defines a *crisis* as a situation which

1. presents a *threat* to one or more important goals of the organization,
2. permits only a very *short decision time* before situation changes significantly, and
3. involves novel or unanticipated events which *surprise* the decision-makers.

Threat or risk to the organization plays a central role in domains such as international politics, corporate management, and military operations. In each case, the situational context is dynamic and complex. The normal states of these environments range from moderately stochastic to indeterminate. The human-computer decision support systems designed to cope with normal operations also must support rapid response to unanticipated events. The organizational, individual, and task implications of crisis operations are discussed further in Sections 2.3.4-6.

2.3.4 Organizational/Doctrinal Profile

As the situational context forms the external environment for decision-making, the structure and goals of the organization provide the internal environment. Systems designed to support decision-making within organizations must take into account not only the hardware, software, and communications architectures with which they cooperate, but also the structure of the human organization in which they function. This involves understanding the organizational culture and how it directly, or indirectly, impacts and is impacted by the individual decision-makers, their tasks and the situational context.[2] Organizational doctrine, whether implicitly or explicitly communicated to the decision-maker, provides not only procedural guidelines for structured tasks, but also a conceptual view and global goals which must be considered. Finally, in the course of evolutionary design and implementation, the designer must be sensitive to the re-definition effects of the new system on the organization and its doctrine. This subsection presents methods for profiling organizations and modeling the relationship of the organization to the other dimensions of HCI design.

Methods for Profiling Organizations

French and Bell (1973) present a hierarchical framework for developing an understanding of organizational functioning based upon information regarding organizational culture, climate, processes, and goals. The framework permits study of the organization as a whole and provides methods for examining and relating the subsystems, teams, and individual functional roles. At each level in the hierarchy, the analyst may select from a range of knowledge elicitation techniques to characterize activities and model the relationship of that level to rest of the organization.

At the top level of the hierarchy, investigation focuses on the organization as an entity with a common mission and power structure. It may also include the relevant external organizations, groups or forces, (e.g., government agencies) and lateral associations that control or interact with the organization. Investigation

---

[2] Tables B-5 and B-6 in Appendix B summarize the key concepts defining the organizational context.

methods include questionnaires, interview, focus groups, and examination of organizational documents (e.g., policies, standards, etc.). In addition to these methods, review of organizational "biographies" or histories of the development and activities of an organization provide insight into the organizational culture (Salama, 1992).

Questions on culture, climate, and attitudes also are relevant at the team or group level of the hierarchy. In addition, the analysis seeks to discover answers to such questions as:

- What are major problems of this group or team?
- How can team effectiveness be improved?
- How well do the member/leader relationships work?
- How does the team relate to organizational goals? Do members understand this relationship?
- How well are team resources employed?

Individual interviews, using techniques such as concept mapping, followed by group review and discussion aid in identifying and refining models of team/group functioning (Klein, 1993b). The models developed may be used in conjunction with more detailed cognitive task analysis to link team structure and function to the specifics of the task and environment.

The function role level of the hierarchy presents the organizational slant on the individual decision-maker and, thus, overlaps considerably with the user profile. Analysis at this level focuses on the set of behaviors associated with an individual position (e.g., leadership roles, functional responsibilities, communication behaviors). The goal of the analysis is to identify the functional roles that impact the performance of the human-computer decision process that the HCI design must support. In this way, the organizational model supports and focuses development of the user profile. Knowledge acquisition methods applicable in this context include field observation, interviews, and other role analysis techniques. Information gathered through these techniques may be used to annotate the organization's job descriptions and "wiring diagrams" of control structures.

The models developed to describe organizational structure and functioning assume additional meaning when augmented by an understanding of the organizational culture. Robbins (1990) identifies 10 dimensions that define organizational culture. These include

- ***Structural Features*** - control, integration, interaction patterns, and rewards;
- ***Management Characteristics*** - direction, support;
- ***Organization Responses*** - conflict tolerance, risk tolerance; and
- ***Individual Characteristics*** - initiative, identification

A strong organizational culture communicates the organization's model of appropriate behaviors to the individual members and increases their identification with the organization. An organization is said to have a strong culture when the core values of the organization are clearly understood, intensely held and widely shared. The resulting unit cohesion prevents breakdowns in procedures in high-stress, crisis situations and is critical for effective performance. For this reason, technologies introduced into a decision organization must facilitate and not interrupt the flow of communication and interaction that supports team cohesion. A strong organizational culture can also have negative effects on decision-making, such as the social pressure for uniformity and failure to question weak arguments common in "groupthink" situations (Janis, 1972).

The concepts of "collective cognition" and the "collective mind" have been proposed to describe the purposeful interaction characterizing team performance in situations requiring a high level of continuous reliability (Weick & Roberts, 1993). The collective mind is evidenced by the manner in which the team members structure and coordinate their actions with respect to a shared mental model of the system. Weick and Roberts' research examined the effects of variations in the individual models and coordination of actions in aircraft carrier flight deck operations. As team members increased the conscious interrelating of their actions within the system they improved their comprehension of unfolding events and reduced the incidence of error. The researchers present a model of collective cognition that relates actions (contributions), the shared mental model (representation), and the coordination of actions within the system (subordination). In related research, Schneider and Angelman (1993) investigated collective cognition in organizations and proposed a cognitive framework based on structure, process and style that is applicable to the individual, group, and organizational levels of analysis.

Examining the formal and informal lines of communication in an organization provides additional information on the means by which control is exercised in an organization. Harrison (1985) discovered that patterns of interaction defined through communication between the hierarchical levels of an organization establish a shared understanding about levels of influence in decision-making processes and how such influence may be exercised. Moreover, the definition of participation through interaction dominated the perceptions of subordinates, regardless of the management style reported by their superiors. The results indicate the importance of actively supporting interaction between levels of the organization where decision-making effectiveness depends upon intra-unit participation.

Organizational Responses to Situational Contexts

Organizations and systems must be designed for effective response in both routine operating conditions and problem situations (Meister, 1991). Organizations develop routine (or standard) operating procedures to guide responses in relatively stable, predictable environments. Although specific tasks may involve some risks, there is usually low threat and adequate response time. In this context, decision-makers respond to problems arising in their sphere of responsibility according to specific guidance from superior authority. These procedures permit a high degree of control and consistency across all organizational levels to ensure organizational objectives are met. The longer decision horizons permit subordinate decision-makers to defer responses when situations exceed the scope of their responsibility. The reduced threat allows decision-makers to reduce their workload through the use of various cognitive short-cuts, or heuristics. Janis (1989) suggests that the cognitive short-cuts used in routine decision-making provide more efficient responses than the conscious pursuit of more precise decisions.

Crisis conditions trigger shifts in organizational communication and control patterns (Hermann, 1972, Meister, 1991). Organizations designed to operate effectively in dynamic, high threat environments must adapt rapidly to crisis conditions and novel situations. Communication delays may impair information gathering and decision implementation. For this reason, decision-makers must respond to novel problems arising in their sphere of responsibility during a crisis with only general guidance from superior authority. There is some evidence that more loosely coupled organizational structures with built in redundancy and informal interaction are necessary to respond effectively in complex, dynamic, high threat environments (Pew, 1988). With training and experience in crisis operations, decision-makers gain experiences to develop a wide range of creative responses; however, their focus on the immediate problem may result in a satisficing response that does not organizational objectives.

Hermann (1972) describes the effects of crisis situations on three organizational dimensions: leadership and control, communication, and decision-making. During a crisis, the active decision-makers are reduced to a core team. The leaders' attitudes toward rank and authority are critical determinants of subordinates' willingness to raise issues that appear to challenge the prevailing hypothesis. Conversely, weak or inexperienced leaders may be influenced by subordinates to make incorrect decisions (Janis, 1989). In crisis operations, there is a marked increase in communication with internal and external agencies. The increased intra-team communication may lead to a general air of confusion (and potentially panic) and increase the impulse to action.

When routine operations constitute the majority of organizational experience, decision-makers have little opportunity to develop a wide range of responses and may be ill-prepared for sudden shifts in the environment. This can have disastrous effects for response coordination. For example, Helmreich (1988) cites NASA and National Transportation Safety Board studies implicating crew coordination in more than 70% of aircraft crashes. Often such cases involved "a minor malfunction or simple error or erroneous assumption compounded by inattention or incorrect decision by the team into a non-recoverable crisis" (Helmreich, 1988, p. 3). Helmreich cites miscommunication (both human-human and human-machine), poor resource use, and inadequate situation assessment as the major contributing factors to the resulting failures.

Designers are rarely able to observe the functioning of organizations during crisis or intense periods of activity. Research indicates that organizational performance during crisis operations may be enhanced through aiding designs that support improved situation assessment and facilitate communication based upon shared mental models (Orasanu & Salas, 1993). The organizational models developed to guide HCI design should explore the decision aiding requirements associated with both crisis and routine operations. The knowledge acquired through these models is used to determine appropriate human-machine task allocation, design information presentation and develop interaction routines. The organizational models also provide structures to link user and task profiles.

2.3.5 User/Decision-Maker Profile

The system users' functional roles within the organization are often developed in conjunction with the profile of the organization. The HCI designer also needs to develop a profile of typical users' knowledge and experience.[3] In certain organizations (e.g., military units), this information may be assumed in part by the functional definition of the position. For example, an aircraft commander may be assumed to have a minimum number of flying hours, to have completed specific training, and passed certain qualifying examinations. The HCI designer also needs information that may not be assumed automatically from job descriptions. To design the information presentation and interaction routines that coordinate the performance of human-computer cooperative decision-making, the HCI designer must develop a profile of the user's knowledge of the domain, the task(s), and the systems involved.

Dreyfus and Dreyfus (1986) identify six levels of knowledge that a decision-maker may progress through in developing expertise. These levels (novice, advanced beginner, competent, proficient, expert, and master) provide a more detailed picture of the role of expertise in cognitive tasks. Intended as an aid in the design of training, the Dreyfus model describes the differences knowledge/skill levels make in the mental functions employed in decision-making tasks and the mental attributes of the decision-maker. The mental functions involved in decision-making tasks include

- differences in ability to recognize similarity in environmental and task features;

---

[3] Tables B-7 - 9 in Appendix B summarize the key concepts defining the user profile.

- differences in the way task components are conceptualized and recognized; and
- differences in the decision strategies employed.

The ability to make similarity judgments is essential for rapid recognition of prototypical situations and analogical reasoning for unfamiliar situations (Beach, 1992; Klein, 1993a). Depending upon level of expertise, the task or situation is either perceived as decomposed attributes (lower levels) or as a whole (higher levels). Expertise also factors in the decision strategy employed. Lower levels of expertise usually require analytical strategies to manage the problem perceived as parameters or attributes. The wholistic models that characterize higher levels of expertise facilitate intuitive strategies. Hammond (1993) also discusses strategy selection based upon the attributes of the task and task situation. Clearly, these models interact to address the combination of factors that determine decision strategy.

Several resources are available to guide the system designer in modeling human users (c.f., Andriole, 1986; Meister, 1991; Senders & Moray, 1991). This section presents simple taxonomic definitions to characterize user knowledge in each dimension as low, medium, and high. These levels represent the continuum of knowledge and experience that usually exists a mixture of expertise -- deep in some areas and broad in others. Each knowledge level is discussed with respect to the potential impacts on decision-making.

Impacts of the Decision-Maker's Domain Knowledge

The decision-maker's domain knowledge is one of the primary resources used to interpret available information during situation assessment. When the decision-maker's domain knowledge is low, they have limited, fragmented models of the domain. This is generally the case with persons who are relatively new to an organization. With only fragmentary domain models, decision-makers have very limited ability to recognize prototypical situations or interpret novel situations. As a result, their framework for structuring response goals is also limited.

A moderate level of domain knowledge developed through training and some experience provides decision-makers with domain models that are largely situation-oriented. At this level, decision-makers recognize some prototypical situations and can use reasoning to respond to unfamiliar situations. Goal structuring at the moderate knowledge level is primarily defined by learned procedures and situational models (Dreyfus & Dreyfus, 1986).

At the highest level of domain knowledge, the decision-maker is a domain "expert" with a wholistic model of the domain. As described in Klein's (1993a) RPD model, the domain expert rapidly recognizes prototypical situations and can intuitively interpret novel situations based on similarities to other prototypical situations. Goals are structured using a robust framework based on the wholistic domain models and an understanding of doctrine.

Impacts of the Decision-Maker's Functional Task Knowledge

The decision-maker's knowledge of the specific functional tasks to be performed interacts with domain knowledge, but is often a very different level. For example, a decision-maker may have considerable knowledge and experience with the situational contexts that characterize the domain, but may have never performed the specific tasks now assigned. In such cases, the decision-maker may understand intuitively *what* must be done to accomplish a goal, but not know *how* to do it. When the typical decision-makers' task knowledge is low, they often cannot distinguish between relevant and irrelevant information needed to perform the task. This lack of knowledge increases their cognitive workload. As described above, the decision-maker with low task knowledge may be unable to generate and evaluate an adequate response.

A moderate level of task knowledge supports task performance based on the decision-makers' facility with learned procedures. This knowledge permits the decision-maker to trade off performance quality in order to maintain a reasonable workload and still attain the desired goal. Moderate task knowledge is adequate for all routine operations and some novel situations.

At the highest levels of task knowledge, the decision-maker demonstrates flexible, intuitive task performance. Depending upon the level of their domain knowledge, the decision-makers can rapidly recognize prototypical situations and adapt their task performance in response. Their knowledge of the task parameters allows them to intuitively interpret task outcomes in novel situations.

Impacts of the Decision-Maker's System Interaction Knowledge

Low system knowledge and how to interact with it can take several forms. For example, it is often the case that a decision-maker has knowledge and experience with the domain and functional tasks, but has had little or no experience using computer-based supports. The decision-maker may only have had occasion to use a few system functions while remaining largely ignorant of its other capabilities. The novice user is just beginning to use the system and must have help to accomplish most tasks. All of these knowledge levels have in common a limited, often fragmented, knowledge of system operation. As a result, the user usually has an insufficient mental model of the system and may be confused by errors. The resulting increase in cognitive workload may greatly impair performance in tasks at which the decision-maker otherwise has proficiency.

The competent user has a moderate knowledge of system functions and the interaction routines required to exercise those functions. The user understands the operation of commonly used system features and can operate the interface to accomplish the required tasks. The competent user's mental model of the system provides an adequate foundation to allow them to learn from operational errors.

The master, or "power" user, has a strong, accurate mental model of the systemic relationships between themselves, the machine, and the tasks they each perform. This system model permits them to coordinate fluid operation of the interface such that the interface operation tasks are "transparent." The user is, thus, freed from the additional cognitive load associated with interface operation and is able to focus directly upon the functional tasks at hand. This level of facility is critical in situations where tasks must be performed rapidly under pressure.

When decision aids play a crucial role in the organization's mission, the information presentation and interaction routines selected must support the anticipated variation in user knowledge across all three dimensions. Where performance reliability is critical, the HCI design must make up the deficit in the user's system knowledge. Depending upon technological feasibility and the goals set for the system, it may also attempt to address deficit knowledge of the domain and tasks. Finally, the HCI design should provide the means for the decision-makers to extend their knowledge and improve their performance.

2.3.6 Functional Task Profile

Modeling Tasks to Determine Requirements

From the user/decision-maker's perspective, the functional tasks encompass the activities the human decision-maker performs to fulfill his role in supporting the organization's mission. Functional tasks include not only the human-machine cooperative tasks and decision-making activities, but also human-human communication activities. These tasks are separate from the system operation (user-machine interaction) tasks that constitute the focus of most traditional human factors engineering. For example, an air traffic controller has functional tasks which include using computer-based support to track aircraft inflight and on the ground, making decisions about control options, and communicating directly with the aircraft personnel. Each of these broad categories of functional tasks must be considered in the development of the computer system that supports the controller.

The early problem definition models provide an initial framework for task definition. During the requirements modeling phase, tasks are iteratively defined using a combination of top-down and bottom-up analysis methods. Andriole (1986) and Ehrhart (1993) describe a variety of task analysis methods useful for investigating both the functional and the interface operation tasks in decision aiding systems. Other resources describe techniques for capturing and modeling the cognitive aspects of decision-making (c.f., Kaempf *et al*, 1992; Klein, 1993a; Klinger *et al*, 1993; Zachary, 1988). Task profiling and requirements identification activities focus on four areas:

- identification and modeling of the sequencing and dynamics of the tasks;
- identification and characterization of decision-critical information regarding the situation elements external to the system (support systems, physical environment, threats, etc.);
- identification of the ways that users interact with all of this information to explore situations, develop hypotheses, generate options, make choices, and implement their decisions; and
- identification of the information presentation and interaction requirements of the alternative analytical methods proposed to support tasks and decision processes.

The remainder of this section discusses the general characteristics of the functional tasks involved in human-computer cooperation and follows with the cognitive characteristics of decision-making tasks.[4]

---

[4] Tables B-10 and 11 in Appendix B summarize the key concepts defining the functional tasks. Table B-12 summarizes the key concepts defining decision tasks.

Characterizing the Key Variables: Inputs, Outputs, and Feedback

One of the principal goals of the task analysis models is to identify and characterize the key variables in the task inputs, outputs, and feedback that define the tasks and affect task performance. The characteristics of these variables and their interrelationship has implications for task allocation, flow of control, information presentation and interaction design, as well as hardware, software, and communications requirements. Meister (1991) lists 15 variable characteristics of task inputs, outputs, and feedback:

• **Modality**	written, spoken, visual, aural
• **Structure**	quantitative, qualitative; structured, unstructured
• **Content**	information provided
• **Intensity**	strong, weak; detectability
• **Immediacy**	immediate, delayed, constant
• **Volume**	(relative to the problem) too much, too little, appropriate
• **Duration**	short, long, continuous
• **Uniqueness**	presence or absence of other associated information
• **Specificity**	specific or general with respect to content or source
• **Consistency**	(with other related information)
• **Source**	internal, external; hierarchical level
• **Linearity**	linear, non-linear (relative to source)
• **Dimensions**	uni-dimensional, multi-dimensional
• **Reference**	organization, unit; external
• **Expectation**	consistent/inconsistent with expectations

These definitions subsume such concepts as reliability, certainty, and ambiguity that affect the combination and interpretation of information. As the tasks and their associated variables are identified, the individual variables must be characterized vís-à-vís these various dimensions and related in order to model the dependencies, information flows, etc. The relationships defined then provide building blocks for design of information presentation and interaction design.

Functional Task Characteristics

The HCI designer often begins with information from organizational job descriptions or the functional role models developed in the organizational profile. In addition to the individual task parameters discussed previously, the designer must develop a profile of the overall shape and flow of the task. This profile considers the principal human functions (i.e., discrimination, communication, interpretation, etc.) required to complete the tasks. In addition, the combination, or cumulative effect, of tasks is examined in terms of such factors as complexity, loading, pacing, and criticality. The overall complexity of the human decision-maker/user's tasks is a function of the number of interdependent factors or sub-tasks involved the task. The level of complexity is highly correlated with task difficulty (Meister, 1991). Also related to complexity, the overall task load describes the demands placed on users by the number of concurrent tasks, interactions, sequencing, etc. Meister distinguished task "load" from task "stress" by the absence of an element of fear or anxiety. Finally, both the pacing and criticality of task performance must be understood to assess their impacts on timing, accuracy and precision, prioritization, and attention requirements

In addition to the overall profile of the functional tasks, the designer must also discover and model the relationships between the elemental aspects of tasks (e.g., variables, constants, actions, processes, etc.). A simple system model interrelating task elements allows the designer to categorize each element in terms of whether it is an input to the task, response activity or process, output from the task, or feedback on action(s) taken (Figure 2.8). This broad categorization helps to identify the characteristics of element that are relevant to the task flow and must be defined further.

Output Characteristics

From the system's perspective, the functional tasks are outputs of the human-computer cooperative response process. For this reason, the identification and analysis often begins with desired outputs. One of the first issues to be resolved is what constitutes an output unit. An output unit may be a single task (i.e., the assignment of a single entity to a service unit) or a composite task composed of a number of elements or component tasks (i.e., planning a series of activities for multiple actors). Task volume, or throughput, is measured in the number of output units produced during a period of time. In some cases, the duration of the output unit is also an issue. For example, an operator may have to maintain some signal or machine state for a set period of time or until an appropriate feedback signal is received. In this case, the workload associated with task output is a function of the number of task output units produced during a set time period and the duration the output is maintained.

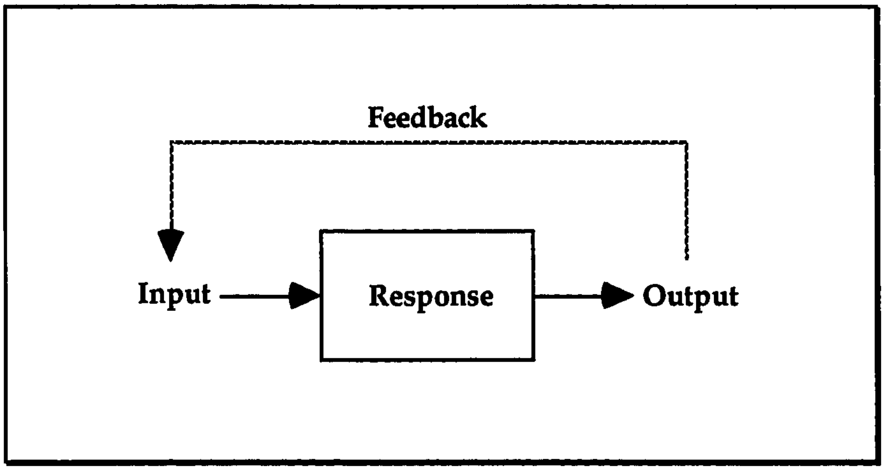


Figure 2.8: Simple System Model of Functional Task Characteristics

The task output characteristics raise several issues for the HCI designer. For example, the output volume required has implications for human attention, workload, and short-term memory capacity that must be considered in human-computer task allocation decisions (Andriole & Ehrhart, 1990; Gardiner & Christie, 1989). The number and format of elements composing the output task unit has implications for the level of detail that must be addressed, manipulated and output. The duration the task output unit must be maintained also impacts attention, memory and workload by limiting resources available to respond to incoming tasks and must be considered in task allocation schemes. Finally, the level of workload associated with output requirements affects not only the task allocation design, but also impacts the cognitive resources required to maintain the level of vigilant performance required.

Response Characteristics

The requirements and characteristics of the user's response are closely related to the output characteristics. For example, task allocation strategies and feedback design depend upon how often the user must respond (response frequency) and how precise the response must be. The difficulty of attaining these response goals becomes a function of the number of component elements incorporated in the task output unit and the output workload. Very low levels for goal attainment difficulty may affect the decision-maker's attention and interest (Meister, 1991). In contrast, very high levels of difficulty may indicate tasks out of the range of human performance. These factors also have emotional consequences in terms of motivation, frustration, and stress. The cognitive demands associated with the content of the decision-making task are discussed in the section on decision task characteristics.

Once the broad tasks are identified, the designer must look at the subtasks or procedures that comprise those tasks. For example, the number and interdependency of the procedural steps required in the response to produce one task output unit also impacts task complexity. The precision required in responding has implications for both the information presentation precision and the means by which the decision-maker/user formulates the response. Tasks and subtasks that must be performed more or less simultaneously create extra demands on attentional focus and cognitive resources (Wickens, 1987). This issue must be addressed in task allocation strategies. Finally, in addition to task precision parameters, the designer needs to take into consideration how closely the user must adhere to prescribed procedures. Tasks requiring absolute adherence to a strict procedure may be candidates for automation. At the very least, the sequencing of valid actions will have to be controlled in the HCI design through the use of constraints and affordances (Norman, 1986).

Input Characteristics

The task input characteristics incorporate the concepts of triggering events (stimuli) and task information that the decision-maker must sense, perceive, attend to, and interpret to generate a response.[5] For instance, the task stimuli or input information may vary over time in a predictable or random fashion. This variation can affect not only stimulus detection, but also the decision-maker's ability to recognize and identify the stimulus. Stimuli with numerous patterns of variation task decision-makers' long-term memory and create additional cognitive workload as they attempt to match features against remembered patterns. The duration of the stimulus relative to the task time and other tasks occurring simultaneously has ramifications for the decision-maker's attentional and short-term memory resources. When the stimulus occurs only briefly or changes while occurring, it may be necessary to store and re-display stimuli for examination. When decision-makers can neither control nor predict the occurrence of stimuli,

---

[5] The decision-making aspects of these task inputs are discussed in the section on decision task characteristics.

they may fail to detect occurrence or recognize significance. Moreover, where task relevant stimuli are mixed with irrelevant stimuli (e.g., a "noisy" environment), the decision-maker may fail to detect the relevant stimuli or mistake irrelevant stimuli as relevant (e.g., "false sensation"). In addition to the added workload, an abundance of irrelevant stimuli can create confusion and seriously degrade performance (Meister, 1991)

Feedback Characteristics

Feedback during task performance informs the decision-maker on the appropriateness and efficacy of the response. In continuous tasks, feedback becomes part of the input for the next response cycle. Feedback on task performance may be characterized in terms of pacing factors such as feedback lag and the ratio of reaction time to feedback lag. When there is no feedback or feedback is greatly delayed, task performance may be impaired (Rasmussen, 1986). In addition, the absence of usable feedback impedes experiential learning (Gardiner & Christie, 1989). Delayed feedback is often mis-interpreted or incorrectly associated with the wrong response causing the decision-maker to construct invalid causal models of the task and domain (Brehmer, 1987; Reason, 1990). When the decision-maker's reaction time must be faster than the feedback returned, the delay in feedback may lead to over-correction in the mistaken belief that the response had no effect. Feedback is also important with respect to the number of subtasks involved in making choices based on feedback on the outcome of the previous response. When feedback is variable in quality or delayed, the effects propagate through a network of dependent choices making the reliability of task performance unpredictable.

Decision Task Characteristics

The cognitive tasks in decision-making comprise the following generic activities:

- ***information processing*** - to collect and organize decision information;
- ***inferencing*** - to interpret information for situation assessment;
- ***judgment*** - to identify a suitable response; and
- ***mental simulation*** - to plan the execution of the chosen response.

Each activity has further cognitive implications in terms of demand or workload (e.g., attention and memory) and potential errors (e.g., biased interpretation or inappropriate heuristic). For example, overloading human attentional and memory resources impacts situational awareness, triggers accuracy/effort tradeoffs, and influences judgment and choice strategies (c.f., Andriole and Adelman, 1989; Janis, 1989; Payne *et al*, 1993; Reason, 1990; Svenson and Maule, 1993). Decision task profiling helps to identify aspects of the task or task sequence that must be supported in the design of information presentation and interaction routines.

Decision tasks may be characterized and modeled using a variety of methods (cf., Andriole & Adelman, 1989; Ehrhart, 1993; Fleishman & Quaintance, 1984). One of the most commonly used general models for decision-making in complex, dynamic situations is Wohl's (1981) Stimulus-Hypothesis-Option-Response (SHOR) model for tactical air combat decision-making (Figure 2.9). SHOR's four generic elements, representing the phases of the decision cycle, are subdivided into the cognitive functions or activities involved in each:

- ***Stimulus*** - the detection/recall, manipulation, display, and storage of the decision data (i.e., situational context and variable inputs).
- ***Hypothesis*** - the creation, evaluation, and selection of alternative perceptions/interpretations of the stimulus.
- ***Option*** - the creation, evaluation, and selection of feasible response alternatives to the hypotheses.
- ***Response*** - the planning, organization, and execution of the selected response option.

Generic Elements	**Functions Required**	**Information Processed**
Stimulus (Data) **S**	**Gather/Detect**	Capabilities, Doctrine; Position, Velocity, Type; Mass, Momentum, Inertia; Relevance and Trustworthiness of Data
	Filter/Correlate	
	Aggregate/Display	
	Store/Recall	
Hypothesis (Perception Alternatives) **H**	**Create**	**COMMANDER'S CATECHISM** Where am I? Where is the enemy? What is he doing? How can I thwart him? How can I do him in? Am I in balance? How long will it take me to ...? How will it look in .... hours? What is the most important thing to do right now? How can I get it done?
	Evaluate	
	Select	
Option (Response Alternatives) **O**	**Create**	
	Evaluate	
	Select	
Response (Action) **R**	**Plan**	**Air Tasking Order:** Who What When Where How The Near-Real-Time Modification/Update
	Organize	
	Execute	

Figure 2.9: The SHOR Model of Tactical Decision Processes (Wohl, 1981)

The SHOR model provides a useful framework for identifying the characteristics, potential sources of error and support requirements associated with the decision tasks at each phase. Using the task characteristics defined in Tables B-12a-d, the designer identifies potential task-related HCI design issues for each of the four decision phases.

Stimulus: Characteristics and Error Sources

The decision stimuli constitute the primary inputs into the hypothesis generation and evaluation for situation assessment. The stimulus phase of decision-making is concerned with initial data gathering and processing. Performance during this phase is determined by the quality of monitoring, focus of attention, and the processing activities (i.e., filtering, aggregation, correlation, etc.) that bring meaning to data gathered. In addition, performance depends upon memory of the evolving context, previous experiences, and training to identify relevance and code stimuli.

In addition to the pacing and volume characteristics of the inputs discussed in the previous section, the data inputs to the decision task must be examined in terms of their impacts on attention, memory, cognitive workload, and information processing. Situational awareness requires varying levels of vigilance depending upon the dynamics of the environment. Therefore, the attentional requirements associated with a decision task may require little active monitoring, monitoring at intervals, or continuous monitoring of the situation. The low monitoring requirements of typically stable or very slowly changing situations may result in poor situational awareness when the stimulus event occurs. When continuous monitoring is required, fatigue can result in loss of attentional focus. Monitoring at set or random intervals incurs additional cognitive workload as the decision-maker may be required to maintain a working memory of the sequence of signals or events monitored in order to create an accurate mental model of the evolving situation. Monitoring at intervals is often involved in divided attention tasks and may require a rapid mental re-orientation each time attention is refocused (Wickens, 1987). Attention is also related to the degree of difficulty in detecting the stimuli. Stimuli that are very difficult to detect, either due to inherent characteristics or the presence of other stimuli, may not attract attention during monitoring. In these cases, stimuli may require machine monitoring for detection or enhancement to facilitate perception or focus attention.

In addition to the cognitive resources demanded by the attention requirements, the pacing and volume of incoming decision data place demands upon the decision-maker's short-term memory. For the HCI designer, these impacts must be evaluated in terms of whether the typical memory demands exceed the capability of proposed users. At the lowest levels, the pace and volume of incoming information are manageable by the average trained user. As the demands are increased, only highly motivated experts can manage the flow of information. The expert uses domain and task knowledge to cluster information in meaningful "chunks" rather than as discrete elements (Badre, 1982). At the highest levels, the volume of information overloads human ability to absorb and manipulate. At this point, machine monitoring and pre-processing is required to aggregate information into more manageable forms.

One of the key issues the HCI designer must examine is the appropriate level of abstraction (i.e., the level of detail) required in the information presentation to permit the decision-maker to effectively interpret the decision data. Rasmussen (1986) categorizes three levels of abstraction for decision inputs: signals, signs, and symbols. Signals are sensed information directly representing time-space data about the environment. Signs are indirect representations of the state of the environment derived from the pattern of physical signals. Signs serve to trigger learned behaviors or rules for response. Symbols are conceptual, rather than physical, structures that represent functional properties and relationships. Signs, or indicators, carry with them a context which triggers not only interpretation, but also expectation. When the situational context differs from the learned context, as in novel situations, it may not be possible to correctly interpret the available information as signs. Symbols represent the more abstract conceptualization of domain relationships necessary in causal reasoning to interpret unfamiliar situations. Forcing decision-makers to work with information at the wrong level of abstraction can either over-burden them with unmanageable detail or provide them insufficient information to adequately assess the situation.

The HCI designer must also determine whether the decision stimuli are primarily quantitative, qualitative, or mixed. Quantifiability is related to extent of structure inherent in the problem (see Section 2.3.3) and has several implications for decision support and HCI design. For example, data in quantitative form conveys an impression of reliability and specificity that may not be warranted by the uncertainty and ambiguity of the variable concerned. Moreover, aggregating numbers derived by estimate rather than direct measurement can result in a compounding of error. While computers can easily manipulate large amounts of disaggregate quantitative data, humans cannot. Information presentation and interaction designs that allocate computation and number storage to the human decision-maker increase cognitive workload and reduce cognitive resources available for performing tasks that are best accomplished by the human agent.

The reliability and representativeness of the input information affects the extent to which the variables may be understood and correctly interpreted. Moreover, when information is incomplete or ambiguous, decision-makers may focus on irrelevant information and inappropriate causal explanations (Reason, 1990). Decision-makers may be unaware that critical information is missing and need reminders or models that call attention to missing, imprecise, or ambiguous values in relevant stimuli. Strategies for analytical support and information presentation require an understanding of which data elements may vary in information reliability and how potential variation may affect interpretation.

Hypothesis: Characteristics and Errors

During the hypothesis phase, the decision-maker seeks to bring an order to the information collected by creating, evaluating and selecting a causal explanation or assessment of the possible situation that would account for the collected data. Several factors characterize the decision tasks during the hypothesis phase. First, the degree to which decisions are made in familiar or unfamiliar conditions affects the reasoning that must be supported and extent to which functions may be automated. For example, routine situations may be handled with procedural reasoning or automated to reduce workload. In contrast, decision-making in highly uncertain environments requires support for interpreting unfamiliar situations. In complex, dynamic environments, human decision-making errors stem from failure to consider processes across time (e.g., evolving trends) and a tendency toward thinking in causal series rather than causal nets (Dörner, 1987).

The decision tasks should also be characterized in terms of the number of feasible hypotheses that commonly may be generated to explain the available information. In well-bounded domains with few possible hypothesis alternatives, situation assessment is usually performed with rule-based, procedural reasoning. Errors in hypothesis evaluation in such instances result from selecting an inappropriate evaluation rule or a flawed evaluation rule (Reason, 1990). In situations where the number of feasible explanations for stimuli may be large, decision-makers may use cognitive short-cuts to rapidly reduce complex hypothetical relationships into loosely integrated general hypotheses. In such cases, the hypotheses may never be adequately integrated for evaluation (Dörner, 1987).

Another dimensional characteristic of the hypothesis phase tasks that must be identified is the time allowed for hypothesis generation, evaluation and selection. Planning and forecasting tasks have longer decision horizons and do not require rapid hypothesis evaluation; however, the delays in feedback can affect the quality of the causal models used to interpret decision inputs. The shortened decision horizon in time-critical tasks increases the effects of decision-maker experience, attention, and workload. The more robust mental models developed with experience increase the decision-maker's ability to focus attention on relevant information, reducing workload to evaluate complex stimuli in shorter periods of time (Shanteau, 1992). Real-time decision-making may require almost instantaneous situation assessment. In addition to experience level and attention focus, decision performance may depend upon vigilance levels maintained and the speed of feedback. (Edland & Svenson, 1993; Janis, 1989).

The stress associated with shorter decision horizons results in general narrowing of perceptual focus ("tunnel vision") or issue fixation, rendering decision makers less capable of dealing with multiple stimuli/issues (Helmreich, 1988; Janis, 1989; Orasanu & Salas, 1993). This tends to result in a decrease in the number of information sources used in situation assessment and the number of alternative courses of action considered. In addition, there is a often a failure to critique the micro-decisions which aggregate to a larger, central decision. The frequency of action or decisions increases as decision-makers feel "impelled" to action.

The nature and amount of inferencing required to interpret situational data impacts the quality of hypothesis evaluation. Presentation contexts, such as luminance relationships, can alter visual perception (Gilchrist, 1990). This effect can be demonstrated by comparing the visual perception of a white piece of paper viewed with a bright light behind it or in front of it. Situational and presentation contexts affects not only the detection of stimuli, but also their cognitive interpretation. In cognitive tasks, the context in which stimuli occur appears to have greater significance than its physical attributes. For example, Lockhead (1992) found context and sequence were the primary factors affecting similarity judgments in recognition and categorization tasks. In other research, Edgell et al (1992) discovered a context effect in the perception of cue salience for probability judgments. The sequence, or presentation order, of decision stimuli has also been found to affect their interpretation in expert situation assessment tasks (Adelman *et al*, 1993). In a series of experimental studies, researchers found that experts constructed different causal explanations for event sequences depending upon presentation order. The explanations provided indicated that the significance experts attached to a particular decision cue differed based upon its sequential context.

The human ability to perceive and interpret information based upon context is an essential strength in situation assessment. When decisions must be made in high threat, dynamic environments, contextual interpretation permits the decision-maker to make accurate assessments intuitively and respond rapidly. Context, however, has also been a factor in mis-interpretation and disastrous decisions. For example, the erroneous shooting of the Iranian Airbus in 1988 by the *USS Vincennes* was, in part, due to the context under which the available information was interpreted (Duffy, 1993; Klein, 1993b). More recently, Pentagon investigations revealed that the April 1994 shooting of two US Army UH-60 Black Hawk helicopters by US Air Force F-15C fighters occurred when the fighter pilots mis-identified the helicopters as Russian-made Hind helicopters flown by the Iraqis. Expectation may have been a contributing factor in the mis-identification. The fighter pilots had not been briefed that allied helicopters would be in the area (Harris, 1994). Other cues (i.e., the negative IFF response and AWACS communication) increased the expectation that the helicopters were either unknown or hostile and may have influenced the visual identification.

Option: Characteristics and Errors

The option phase seeks a feasible response to the hypothesized situation. Several characteristics of tasks during option generation, evaluation and selection bear examination during task modeling. Many of the same factors affecting hypothesis generation and evaluation (e.g., situational context, boundedness, and tractability) also influence the performance of the option phase tasks. The number of potential responses to a situation affects the boundedness of option evaluation. Furthermore, when there are many feasible options to a situation, decision-makers may shift from option to option without sufficient evaluation or attempt to oversimplify (Dörner, 1987; Janis, 1989). Information volume and problem boundedness also affect tractability and may cause the option evaluation task to exceed human manipulation abilities. The goal variability inherent in the environment impacts option evaluation based on the rapidity and predictability of the variation and resulting option conflicts. In multi-stage, evolving decisions, a change in goals may supersede previous sub-choices. Such shifts require rapid re-prioritization and re-evaluation of current options against higher-level goals (Klien, 1993b). Feedback timeliness also becomes more critical as goals shift rapidly.

The difficulty of option evaluation tasks is judged by the extent to which outcome values are well-understood and easy to determine. In bounded and semi-bounded domains with well-understood outcome values, decision-makers may employ rule-based evaluation. Higher levels of evaluation difficulty become less tractable for unaided evaluation. At this point, the decision-making may be unacceptably delayed as decision-makers wrestle with the possible consequences of possible courses of action. Inferencing is required where outcome values are uncertain. In complex environments, the network of uncertainties rapidly becomes intractable for human evaluation, leading decision-makers to simplify with insupportable inferencing leaps (Dörner, 1987; Hogarth, 1987).

Decision-makers may also avoid committing to any option, often waiting to see if changing events force or suggest a choice (Janis, 1989).

Response: Characteristics & Errors

The response phase involves planning, coordination, and execution control required to carry out the course of action option selected. Plans are essentially hypotheses based on a network of causal assumptions about the sequence of steps that will bring about the desired goal. Simple responses involve little or no planning. Skill-based control evokes non-planned reactive responses; rule-based control triggers procedural plans (Rasmussen, 1986). Moderate levels of planning feature manageable levels of effort using ad hoc or pre-packaged plans. Complex responses usually require extensive planning or replanning involving the re-evaluation of goals and adjustment of control structures. Reason (1987) categorizes plan failures as mistakes, that is, errors of intention and suggests three basic sources of planning failures, including

- errors in the working database (i.e., stimulus phase errors)
- errors in the mental operations (i.e., hypothesis and option phase errors); and
- errors in the properties of the schema (i.e., the plan itself).

Reason traces these errors to characteristics of the human planner based upon limits of attention and memory, and powerful urge to accept explanations that bring order to complex, chaotic situations.

The response coordination requirements are determined based on the size, complexity, and dispersion of the network of the agents that must be coordinated. These elements depend upon the organizational factors discussed previously and the time available for response. Coordination tasks are communication intensive with corresponding impacts on support design. In crisis situations, the effectiveness of coordination is dependent upon experience, training, shared task and situational models, and flow of communication.

Execution control is defined in terms of the number and interdependency of the step required in the planned response. As such, control is closely related to coordination. Multi-phased, interdependent responses increase the coordination effort required to track the status of the evolving response. Moreover, the network of dependencies increases the difficulty of tracking all the possible consequences or "ripple effects" of actions taken. If feedback is delayed, it may be associated with the wrong phase and result in confusion and over-correction (Meister, 1991). Finally, the additional cognitive resources (e.g., attention and memory) are demanded to handle the wider range of control and potential goal-shifting in multi-phase responses.

Relating Cognitive Task Characteristics to Task Models

Investigation of the situational, organizational, user, and task dimensions helps to identify the specific aspects of the decision tasks that should be considered in the design of the decision information presentation and interaction routines. Figure 2.10 relates the requirements dimensions (and supporting tables in Appendix B) that combine to define the cognitive task requirements (CTRs). The decision-maker's cognitive tasks emerge as part of describing the sequence of steps involved in performing a task or procedure. As discussed in the previous section, the tasks in decision-making involve generic cognitive functions such as information processing, inferencing, judgment, and mental simulation. Each of these functions is influenced by the situational context, organizational structure and culture, the decision-maker's experience and training, and the inherent features of the task.

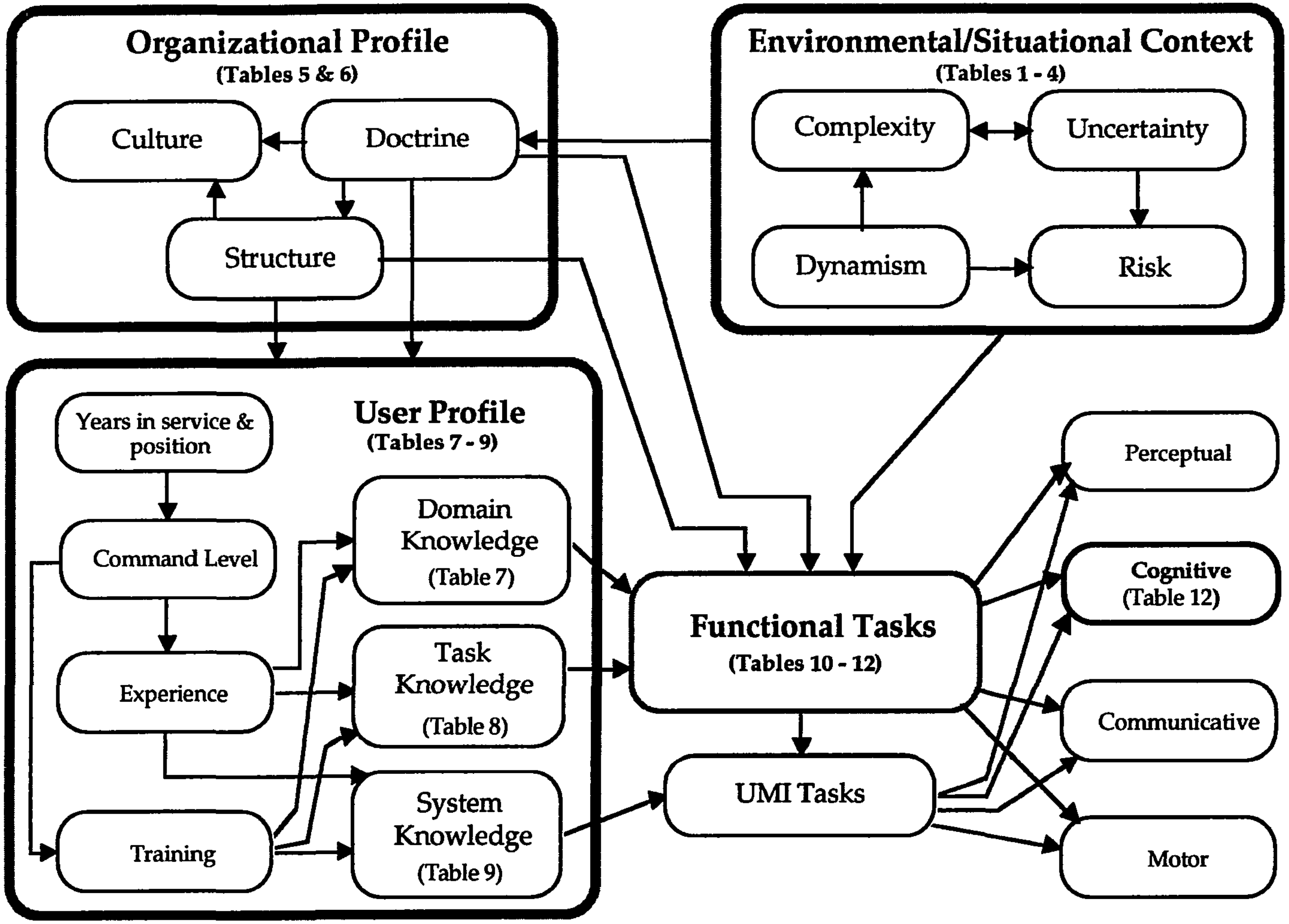


Figure 2.10: Relationships Between the Requirements Dimensions and the Cognitive Task Requirements

As the task models are developed, the designer can begin to explore the cognitive requirements involved in successful task performance. As the designer works through the tables in Appendix B and construct models of the problem domain and tasks, the issues raised may be compiled for later distillation and structuring. Figure 2.11 presents an example of how the CTRs are derived from the issues raised while analyzing the decision task requirements as presented in Table B-12a-d of Appendix B. The tables provide the means to characterize various aspects of the decision domain and tasks in terms of readily observable, broad criteria. Location of the domain and tasks within certain parameters suggests possible sources of cognitive demand and decision-maker error. These potential problems are evaluated in terms of system support and expressed as cognitive task requirements. Appendix F presents an example of a list of issues raised during requirements identification and analysis that is structured to match the tables in Appendix B. Once the common elements are identified, the list is abstracted to generate the CTRs. Section 6.0 in Appendix E summarizes the CTRs identified in through this process.

Figure 2.12 takes the example decision task (Figure 2.6) presented at the beginning of this section and indicates some cognitive support issues that might surface during requirements identification and modeling. For example, if identification of the data inputs to Factor F revealed a variation in the timeliness and reliability of the data, this fact must be considered in presentation of that information to the decision-maker. The reliability will also be a consideration in determining the analytical method used to track and compare the change in Factor F. Additionally, if Factor F is also a multi-dimensional construct (e.g., time, location, and capacity/range), the combination of those dimensions must be presented in a form that is meaningful to the decision-maker and representative of the underlying relationship.

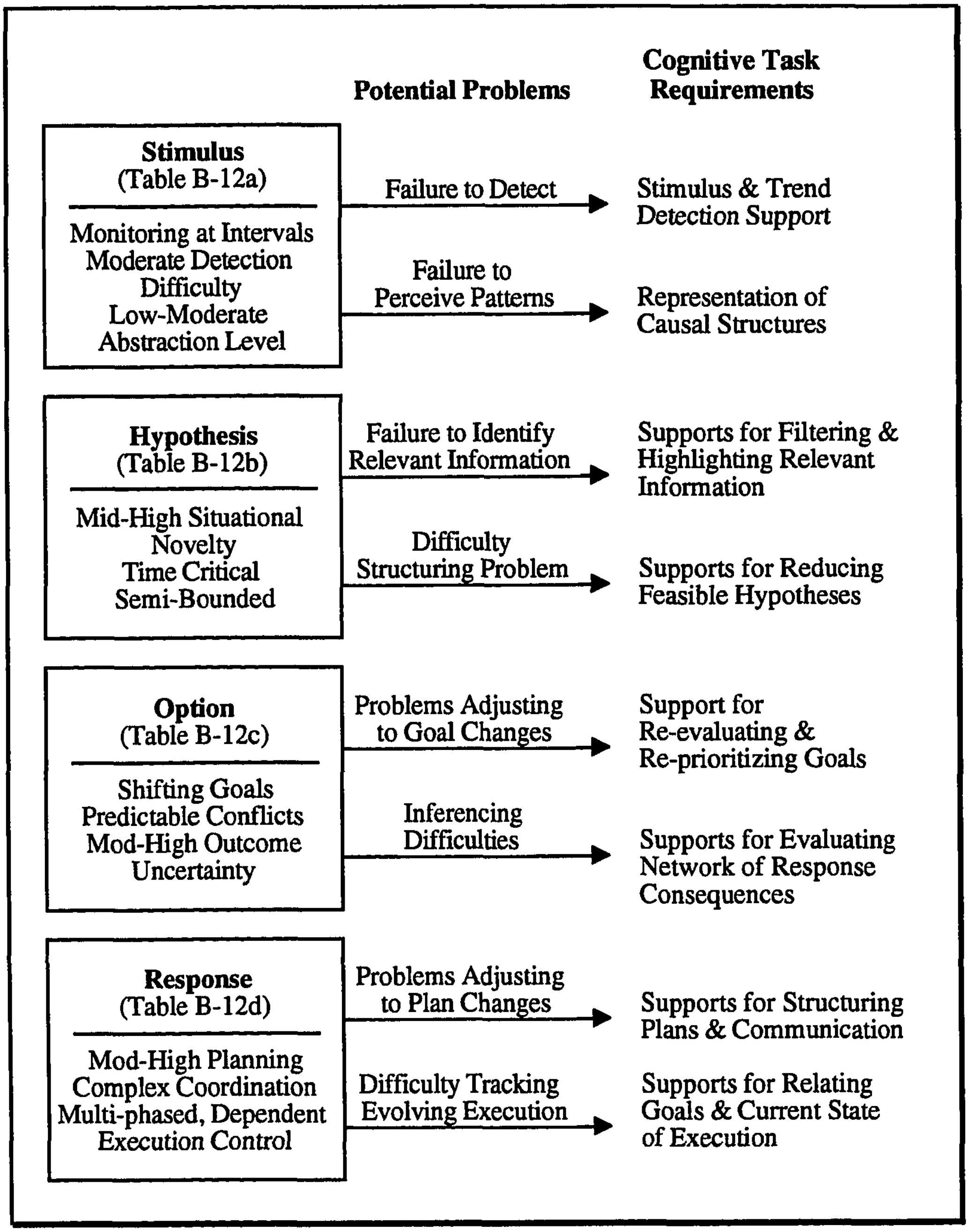


Figure 2.11: Example CTR Inputs Derived from the Decision Task Characteristics

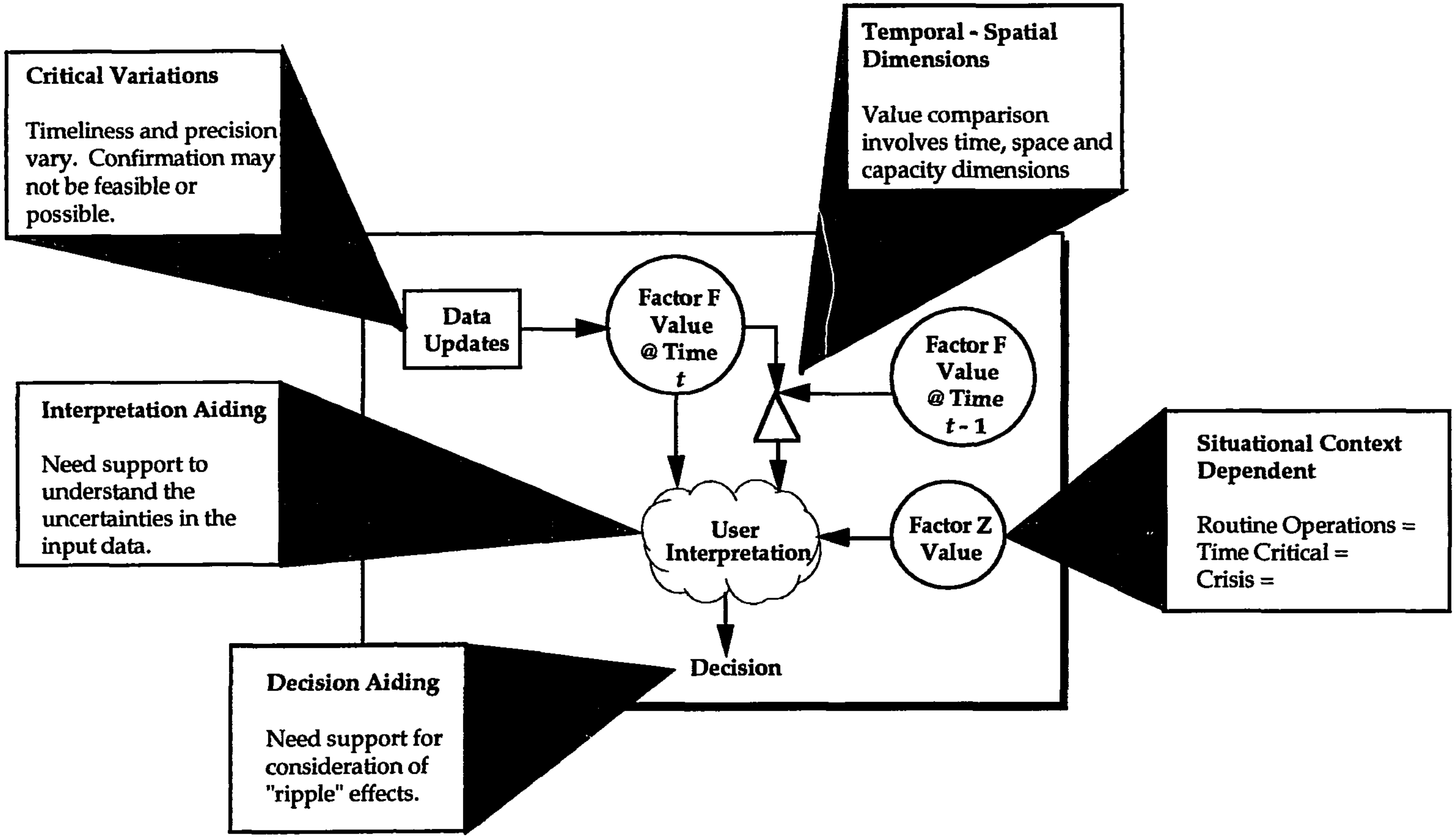


Figure 2.12: Interpretation of Task Characteristics in Example Decision-Making Task Model

Today's HCI designer has a range of software tools that facilitate the creation of task requirement models. For example, modeling software that permits hypermedia links provides the means for "annotating" the basic task models with additional models (i.e., situational context, etc.), text-based descriptions, audio clips from interviews, and even field video of task performance under realistic conditions (Ehrhart and Aiken, 1990). The process of building and reviewing these models helps to identify the cognitive characteristics of each task. These cognitive characteristics, in turn, raise performance and HCI issues that should be included in the requirements specification to assure their inclusion in the design and implementation of the system.

The next section presents suggestions for representing and integrating cognitive task requirements in the requirements specification document.

2.4 Phase Three: Developing an Integrated System Requirements Specification (SRS)

To facilitate reviews, inspections, and later evaluations, the CTRs must be integrated into the system requirements specification (SRS) document. Davis (1993) describes the contents of the software requirements specification (also abbreviated as SRS) as a "complete description of the external behavior of the software system" (Davis, 1993, p177). When effective performance is dependent upon the cooperative interaction between the human user and the software, the external behavior of the cooperative *system* should be represented in the **system requirements specification (SRS).**[6] The purpose of this section is to suggest ways in which the CTRs may be included in the specification of system requirements. As such, it is not an attempt to define a new method or standard for system requirements specification. For this reason, much of Davis' discussion

---

[6] For the purposes of this discussion, SRS refers to this system-level specification.

of software requirements translates directly to the specification of CTRs in system requirements specification.

A clearly and concisely written SRS is a fundamental part of risk management in system development. By Davis' definition, the purpose of the SRS is to support the following activities:

- **communication** between the sponsoring agency, the development teams, and the end-user;
- **evaluation and system testing** at each phase; and
- **planning and control** of system evolution.

Clear communication in requirements specification helps to raise issues of disagreement between the sponsor, end-user and development team during the early phases of development -- rather than during acceptance testing. The system requirements form the core that defines acceptable performance and, thus, must reflect verifiable features and behaviors for evaluation. Finally, the requirements documents provide input into planning and control for project management.

The decision to include a CTR statement in the SRS should be guided by the contribution it makes to the above purposes. Davis (1993) presents 14 attributes of a "well-written SRS." Several of these attributes suggest general heuristics for determining the inclusion and evaluating the representation of a CTR:

- ***Consistent*** - Is the CTR consistent (i.e., non-contradicting or non-conflicting) with other CTRs and system requirements?
- ***Correct, Complete and Unambiguous*** - Does the CTR accurately and unambiguously state the required cognitive support?
- ***Understandable for Interested Parties*** - Will the various decision-makers and participants in the development process (e.g., sponsors,

end-users, designers, etc.) understand the terminology and meaning in the CTR?

- ***Design Independent*** - Is the CTR stated such that it allows the consideration of several alternative design implementations?
- ***Verifiable*** - Does the CTR represent a measurable attribute?
- ***Traced and Traceable*** - Does the CTR trace back to early statements of system requirements (e.g., statement of work, higher-level requirements)? Does the CTR state the requirement such that it can be traced from design back to the SRS?

Given a functional requirement to monitor a situation, the statement of the related CTRs (in bold face type) might take the generic form in Figure 2.13. The first CTR statement indicates that the decision-maker/user must have the ability to examine trends of change across a period of time that may vary depending upon the decision-maker's problem. The next CTR indicates that Factor F is multi-dimensional and lists the relevant dimensions. No specification is made about the nature of the representation. That aspect will be addressed in the context of other requirements during design. The final CTR specifies three attributes that may contribute to the reliability of the data input as Factor F. Note that as currently stated, there is no requirement for the computer to assess reliability. This requirement leaves the burden of judging reliability to the decision-maker, but presents the information required to make the judgment. As such, this CTR specifies the minimum requirement for adequately supporting the decision-maker, but the development participants must be aware that the burden of reliability determination has been assigned to the decision-maker/user.

Not all user requirements may be meaningfully represented in this discrete format. Where cognitive functions may effectively be represented in diagrammatic task models, those models provide annotations to clarify the natural language requirement statements. In addition, certain contextual requirements may be included in the diagrams and narrative descriptions that preface requirements specifications.

The next section presents guidelines for translating the cognitive task requirements into the HCI design concept.

Purpose

This system capability provides the facilities which enable the user to review and monitor data which will facilitate the analysis of the impact of changes in available resources or external environment on the capability to accomplish Task *X*.

Requirements

a. The system shall provide facilities which permit the operator to detect changes in available operational resources which exceed a previously defined threshold.

  - **The system shall provide facilities to display and compare the increase and/or decrease in [Factor *F*] during a user-specified time [*t*.]**

  - **Representation of [Factor F] shall include the following dimensions:**
    - **» [dimension 1]**
    - **» [dimension 2]**
    - **» [dimension 3]**

  - **Representation of [Factor F] shall include indication of the source, time reported, and time received.**

Figure 2.13: Example of a CTR Integrated in a System Requirements Document

2.5 Phase Four: Translating Requirements into an HCI Design Concept

During this phase, the design team begins to match the CTRs to CSE design principles, such as those presented in Gardiner & Christie (1987), Rasmussen *et al* (in press) and other sources. These resources interpret the cognitive demands that characterize certain tasks and task situations in terms of the impacts on information presentation and human-computer interaction. When coupled with basic human factors guidelines for HCI design, the CSE design principles help to identify technological solutions which support the CTRs and conform to the identified hardware and software requirements. For example, selectively focusing attention is a coping strategy invoked when the decision-maker is overwhelmed by large amounts of information. This information processing strategy may be associated with such biases as fixation on one problem element or over-emphasis of cues that support the current hypothesis. The CSE design principles which address "selective attention" include the following:

- Provide reminders of the "larger world" to avoid tunnel vision, and
- Provide means for directing user focus to most relevant information.

The HCI design goals for implementation of these principles might include:

- Provide an overview, or "establishing shot," to expand the decision makers perspective, and
- Exploit common representational analogies (e.g., maps, models, etc.) to highlight the relationships between domain factors.

As new requirements and related design "goals" are identified and understood, they can be integrated into the developing system concept. Rather than occurring in a rigid sequence, this process continues iteratively as requirements surface and prototype concepts are proposed. In this fashion, the prototype HCI design evolves as the incarnation of the designers' hypotheses regarding the decision making activities and interaction requirements.

Figure 2.14 models the formulation of design goals based on informal requirements knowledge (embodied in the situational, organizational, user, and task models), formal requirements specification, and guidance literature (i.e., CSE principals and HCI guidelines). The HCI design concept is a configuration of features including the information presentation methods, interaction routines, and the hardware and software technologies that support them. Each feature must be traceable to the SRS. The specific incarnation of the feature and its configuration in the design should be traceable to the higher-level design goals, principle(s), and guideline(s) that defined or suggested it. This dual traceability ensures that the proposed design adequately meets requirements and helps the design team make better use of HCI technology options available to them.

For purposes of generalization, the discussion of HCI design goals presented here, as well as the principles and guidelines underlying them, is restricted to the higher-level design goals. The design practitioner is directed to the HCI and decision aiding literature for more detailed presentation of

- ***principles and guidelines***
  (Gardiner & Christie, 1987; Rasmussen et al, in press; Shneiderman, 1992; and Smith & Mosier, 1986)
- ***empirical and experimental evaluations***
  (Castellan, 1993; Klinger et al, 1993; Svenson & Maule, 1993;)
- ***theoretical foundations***
  (Card *et al*, 1983; Dreyfus & Dreyfus, 1986; Janis, 1989; Klein *et al*, 1993; Meister, 1991; Norman & Draper, 1986; Rasmussen & Vicente, 1989; Senders & Moray, 1991)

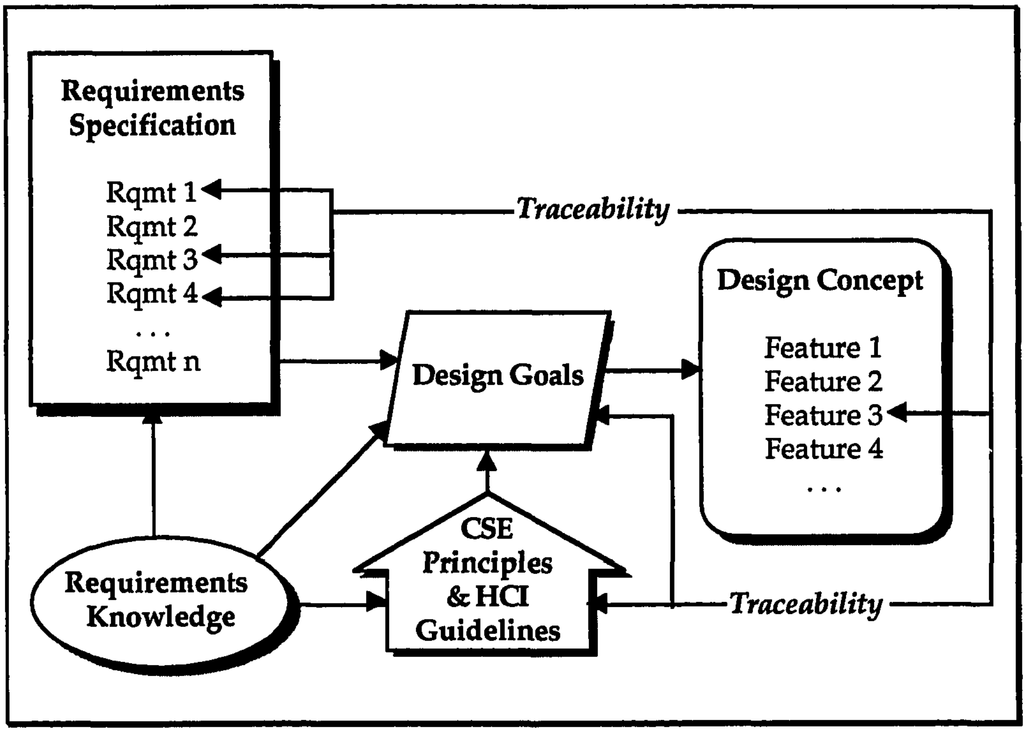


Figure 2.14: Translating Requirements into an HCI Design Concept

The remainder of this section surveys some the CSE principles and HCI design guidance that relates to the situational, organizational, user knowledge, and task characteristics presented in Section 2.3. Each category is discussed in terms of information requirements, support for potential performance errors and possible design goals. This information is also summarized in Tables B-1 through B-12 in Appendix B where the respective characteristics are presented.

2.5.1 Design Goals Associated with Situational/Environmental Context

Vicente and Rasmussen's (1992) Ecological Interface Design (EID) model presents two environment-related design goals based on Rasmussen's (1986) model of cognitive control. First, the interface design should not force the decision-maker to use a higher level of cognitive control than required by the task. Empirical evidence suggests that the skill-based and rule-based levels of cognitive control produce the most efficient response, provided the decision-maker has correctly interpreted the situation. In addition, there is evidence that decision-makers attempt to reduce task demand by relying on the cognitive short-cuts provided by the lower levels of control (c.f., Klein, 1993a; Rasmussen, 1993; Rastegary and Landy, 1993; and Wickens *et al*, 1993). Second, the interface should support all three levels of control (i.e., skill-based, rule-based, and knowledge-based). This goal reflects the decision-maker's requirement to operate in the multiple environments that make up complex domains.

In determinate environments, the principal design goal is providing support for decision-makers to help them rapidly select an effective response to a relatively unchanging and predictable environment (Meister, 1991; Rasmussen, 1986). The limited, highly structured set of cause and effect relationships permits response automation when very rapid response is required. Decision-makers need detailed displays that present specific values for the parameters (e.g., altitude and air speed in aircraft). Where those values must be considered together, the display should either integrate them or present them in sufficiently close proximity that the decision-maker can compare the readings almost simultaneously (Vicente & Rasmussen, 1992). Interaction should be designed to allow the decision-maker to act directly on the display to manipulate the time-space signals.

In moderately stochastic environments, the decision-maker needs to understand the effects of variability in some parameters and the interaction of the parameters. In some cases, the display of some individual parameter values may be integrated into a single display for interpretation as signs rather than as signals. There is empirical evidence that indicates the use of "configural displays" improves performance by allowing decision-makers to extract critical data relationships from both the low-level parameter values and the high-level constraints (Bennett *et al*, 1993). Woods and Roth (1988) indicate the strength in configural displays lies not only in the economy of representation, but in the emergence of certain domain features. It is important, however, that displays representing complex domains not reduce the complexity below the level of the fundamental parameters and their interdependencies. Rasmussen *et al* (in press) refers to this requirement as the "law of requisite variety."

In severely stochastic and indeterminate environments, the HCI design goals focus on providing the means to make most efficient use of resources in a succession of varying, short-term situations. Decision-makers must be able to rapidly develop creative, adaptive responses to effectively exploit opportunities and avoid disasters. This requirement suggests the need for displays that represent the causal relationships and make use of goal-relevant domain models. The representation of causal networks provides externalized mental models that relieve the decision-maker of the cognitively demanding tasks involved in comprehending the causal factors underlying a situation and the network of consequences associated with options (Rasmussen and Vicente, 1989). As such, these displays help to support the mental simulation required for the intuitive response patterns suggested in Klein's RPD model (Klein, 1993a).

Table 2.2 summarizes the HCI design goals related to the situational and environmental contexts which the human-computer cooperative decision system must operate.

Design Goals Summary

- Support all three levels of cognitive control: skill-based, rule-based, and knowledge-based.
- Support skill-based control with displays and interaction methods that allow decision-makers to directly manipulate the signal-level parameters of the problem.
- Support rule-based control with displays that map the structure and constraints of the environment. Model structural relationships and make domain variables salient through design and highlighting.
- Support knowledge-based (or model-based) control with domain models that help to relate the problem parameters to goals. Model causal relationships and make goal-relevant information salient through design and highlighting.

Table 2.2: Summary of Design Goals Related to Situational/Environmental Context

2.5.2 Design Goals Associated with Organizational Contexts

Response selection and coordination within an organizational context involves synchronizing multiple perspectives, synthesizing intra-organizational information, and recognizing relevant patterns in evolving situations to formulate an appropriate response. The design goals associated with the organizational context focus on the responding to the interdependencies of the organizational structure, facilitating communication, incorporating accepted doctrine, and supporting the shared mental models required for effective organizational response.

In organizations that feature complex, interdependent structures, the performance of one unit or subsystem affects the performance of the others. The extent of this effect may range from enhancing or degrading the other function's performance to a tightly-coupled relationship where one function cannot be performed if the other fails. In either case, the decision-makers responsible for the performance of a function within an interdependent structure must maintain some awareness of the organizational functions that support their functional responsibility, as well as the organizational functions that are affected by their decisions. Decision-makers must consider these causal factors in contexts where knowledge-based control is used to adapt to complex, dynamic environments. Depending upon the tasks supported and degree of interdependence within the organization, HCI design goals for organizational structure may include models that relate the dependent network of supporting functions for diagnostic reasoning and situational awareness. In addition, causal models can provide reminders of the potential consequences of decisions for other organizational functions. Finally, models may present the flow of coordination and control involved in implementing decisions within the organization.

The shifts in organizational response during crisis may also require some attention in the HCI design. For example, if decision-making is performed in a distributed environment, the decision-maker may have to cope with failure of communication links that provide updates to critical information. The design of information presentation must provide indications of the data elements affected. The interaction design may include methods for reorganizing the display of information given the changes in data reliability. The HCI design may also have to accommodate shifts in decision-making autonomy under crisis conditions. In these cases, the standard operating procedures and channels of authorization may be replaced by a set of high-level goals and constraints (e.g., military rules of engagement) to permit faster, semi-autonomous responses. Based upon the information gathered in requirements, the information presentation design and interaction control should be adaptable to these conditions.

Wellens (1993) presents an information-processing model for multi-person and human-machine decision-making in a distributed decision-making environment that addresses some of the problems of communication design. Wellens' model incorporates the concept of communication bandwidth (i.e., the degree of richness in communication) associated with the modes of interaction and communication supported. For example, video-conferencing provides all the cognitive content of face-to-face discussion, but "filters" out some of the behavioral aspects that are sometimes counter-productive. This "filtering" is not function of the electronic medium, rather it is due to the participants' tendency to focus on rational presentation of factual information without additional emotional behaviors. Despite the intuitive appeal of increasing communication bandwidth, Wellen's experimental research with dynamic situational awareness in team decision-making indicated increases in information richness were not always associated with improved situational awareness. This result seems to be due largely to the time pressures and the additional filtering required in an information rich medium.

The HCI design goals for supporting communication in the organizational context should evolve out of an understanding of who must share information, what information must be shared and how it must be communicated. Within this high-level construct, information interaction design concepts should strive to maintain an appropriate distance and directness in the communication between members of the team or organization. As such, the design should facilitate the integration of decision-makers who must cooperate and not interfere with their cooperative tasks.

Much of the strength in shared mental models appears to be task, training, and communication dependent. Rouse *et al* (1992) state that the current empirical evidence is insufficient to form a coherent theory of team-based design. In fact, there seems to be some evidence that technology interferes with shared mental models. Duffy (1993) cites the loss of "backchannel communication" as a potential negative effect of introducing technology in team processes. The communication that occurs in the background of the primary communication provides team members the opportunity to question, clarify, and confirm their understanding of the situation. This secondary communication is a critical part of avoiding errors due to miscommunication. For example, the investigation of the Black Hawk helicopter shooting indicated that some of the members of the AWACS team knew before the shooting that the helicopters were US Army Black Hawks, but the information did not get communicated to the pilots of the F-15Cs (Harris, 1994).

Group or team situational awareness is "the sharing of a common perspective between two or more individuals regarding current environmental events, their meaning and projected future status" (Wellens, 1993, p. 272). HCI designs to support shared mental models should incorporate not only the advantages of multiple perspectives, but also the power of shared knowledge and training. This shared knowledge includes doctrinal concepts and common representations of both abstract and concrete organizational information (Kahan et al, 1989).

Table 2.3 summarizes the HCI design goals associated with the organizational context.

Guideline Summary

- Provide models of the interdependencies in the organization to aid the decision-maker in assessing the causes of situations and effects of choices.
- Provide the means for decision-makers to adapt to the shift in organizational response during crisis situations.
- Encourage consideration of organizational doctrine through the use of goal- and constraint-based displays.
- Facilitate all necessary and useful communication between decision participants with information display and interaction concepts that support team interaction.
- Support sharing of team or unit mental models to foster effective task coordination.

Table 2.3: Summary of Design Guides Related to Organizational Context

2.5.3 Design Goals Associated with Decision-Makers Profile

The decision-maker profile identifies the predicted levels of knowledge, experience, and training that the decision-makers/users are likely to have with respect to three knowledge areas: the domain, the functional tasks, and the operation of the system. The effects of this knowledge generally conform to models of the beginner (low level), competent practitioner (moderate level), and the expert (high level). Individual system users typically demonstrate a range of competency across the three knowledge areas. The three knowledge levels have a number of common features, regardless of the area of knowledge involved. As with cognitive control, the predicted knowledge levels of the prototypical user must be supported for each area. Each knowledge level is discussed below with design goals for each area of knowledge. The design goals were synthesized from Dreyfus and Dreyfus (1986), Rasmussen (1986), Rasmussen *et al* (in press), and Senders and Moray (1991).

Low Levels of Expertise

At the lowest level of expertise, the decision-maker/user may not recognize critical cues regarding the situation, task or system state. In addition, the decision-maker usually has only limited ability to reason about the cues provided. In novel situations this limitation may induce confusion and error. The beginner often lacks confidence and may be slower to respond and reluctant to commit to action. Finally, lower expertise is associated with a limited goal framework that increases the probability of errors of intent.

Where domain knowledge is low, decision-makers benefit from displays formatted as accepted domain models to present situational information in context and map causal relationships. Constraints, supports and reminders help to guide domain understanding and increase confidence in situation assessment. In addition, templates of prototypical domain constructs with relevant cues highlighted can assist the decision-maker in making comparisons and developing responses in novel situations.

Low task knowledge often results in an inability to handle shorter decision horizons and heavy information loads. Additional time may be lost reviewing irrelevant information or inappropriate options. As a result, the beginner has difficulty maintaining performance quality under increased task workload. Lower levels of task knowledge are characterized by limited response option generation and evaluation capabilities. Finally, the beginner has difficulty prioritizing tasks. Display and interaction supports for functional tasks are similar to those discussed for low domain knowledge. To support the beginner in developing task knowledge, the HCI design should allow the user to query constraints and affordances built into the task models. Automation strategies should be explored to relieve the beginning decision-maker from excessive cognitive workload. When feasible, adaptive "intelligent" decision aids may be appropriate to filter displays and propose options. Where this type of aiding is infeasible, organizational structures may provide the same kinds of error trapping, error flagging, and redundancy afforded in machine design.

Low system knowledge is addressed in most fundamental guidance for HCI design (c.f., Bailey, 1989; Shneiderman, 1992). Several general guidelines apply to help reduce errors and foster system learning. First, the information presentation design should provide overview screens to help users develop a mental model of the system resources available and understand where they are in a process. Moreover, the human-machine communication should make the current state of the system implicit and make the available options visible. The interaction design should include built-in constraints to prevent an unrecoverable error, alert the user to nature of their error and their current response options. Finally, Norman (1986) encourages designers to make use of natural or domain knowledge in the interaction symbology to allow the user to interact with the task in the most familiar terms.

Moderate Levels of Expertise

Moderate levels of expertise lead to performance errors based on misinterpretation of cues due to limits of the decision-maker/user's domain, task, or system models. Alternatively, errors can occur when the decision-maker fixates on the most available models. Moderately experienced decision-makers have limited ability to resolve conflicts between multiple models. Finally, moderate expertise is characterized by a reliance on learned procedures and a limited ability to reason at higher levels of abstraction in unfamiliar situations.

Moderate domain knowledge may be supplemented with displays formatted as accepted domain models to present situational information in context and map causal relationships. The HCI design should support construction of more robust mental models by providing the option to view deeper levels of explanation. Since decision-makers may fail to recognize the degree and impacts of uncertainty in situational cues, displays and interaction routines are required that make the sources and extent of domain uncertainty explicit.

Moderate knowledge of the functional task requires some of the same support described for lower knowledge levels. For example, the decision-maker's task knowledge may not be sufficiently robust to understand the effects of subtask uncertainty. Displays and interaction design should help the decision-maker to understand the source of uncertainty and explore the potential effects on task performance. Moderate levels of task knowledge also benefit from designs that make task constraints and affordances visible. In high information volume situations, the decision-maker may not have adequate schema to distinguish relevant information. The HCI design should provide goal- or decision-oriented displays to focus attention on relevant information and provide natural constraints for error control.

Moderate system knowledge is characterized by response mode errors based on incorrect assumptions about the current system state. For this reason, system state, available options and similar information should be visible or available on demand. It is also beneficial to minimize the use of similar interaction sequences that vary in effect given different operational modes. Moderate levels of system operation expertise may not provide sufficient procedural information to respond to unexpected system behavior. In addition, the competent user may become lost in complex, linked sequences of displays. Overview displays and interaction routines that help the user to trace recent steps help the user maintain orientation (Woods, 1984). HCI designs for moderate system operation knowledge should still facilitate error recovery with "undo" commands and similar recovery devices. Finally, the design should feature multiple levels of help to allow the user to select the depth of information desired.

High Levels of Expertise

The highest levels of expertise continue to feature errors in the selection and interpretation of information and judgments regarding appropriate responses. Although decision-makers have expert levels of domain knowledge, they may exhibit inconsistencies in combining situational cues. In addition, the multiple models in their repertoire may compete, with selection triggered by availability rather than reasoned choice. Experts benefit from the option to use domain model displays or customize displays and interaction routines to match their mental models. As with the moderately experienced decision-maker, experts require HCI designs that support the continued development of mental models and provide the option to view deeper levels of explanation. Expert decision-makers may display over-confidence in their situational interpretation or response choice. Displays that make explicit the sources and extent of domain uncertainty continue to be useful at this level.

High task knowledge is also plagued with over-confidence. This stems in part from an insensitivity to the potential for aggregated errors in subtasks (microdecisions) performed in multistage decisions and a failure to revise decisions in light of new information. For this reason, experts continue to benefit from constraint representations for error control and the option to use supports and reminders during situation assessment. These may be provided in goal-oriented displays or displays and interaction routines user-customized to match their mental models. These displays also help to promote understanding the causal network of contributing causes and consequences of action. Finally, expert decision-makers' difficulties adequately considering domain uncertainties may be reduced with displays that make the sources and extent of uncertainty in key variables explicit.

The decision-makers with high-levels of expertise in system operation can still be confounded by illogical HCI designs. In general, the rules for consistent design of information presentation and interaction routines discussed for the lower levels of expertise apply to the expert. Several additional considerations apply primarily to the higher levels of system operation knowledge. For example, expert system operators are usually very intolerant of being forced to use lengthy procedures to accomplish a simple task. The HCI design should allow the user to tailor the interface to optimize for best performance. However, when a user reaches expert levels in system operation, their ability to bypass some operational sequences may result in unintended actions. For this reason, experts also benefit from the error-tolerant design guidelines suggested for lower levels

Table 2.4 summarizes the HCI design goals associated with the decision-maker/user's knowledge and experience in the domain, tasks, and system operation.

Design Goals Summary

- Provide support for predicted levels of decision-maker/user knowledge and experience with domain, functional tasks, and system operation.
- Provide less experienced decision-maker/users with reminders to support performance, constraints and recovery routines to prevent serious errors, and embedded models to promote learning.
- Provide moderately experienced decision-maker/users with goal- or decision-oriented displays to aid in reasoning with multiple models.
- Provide highly experienced decision-maker/users with ability to take short-cuts and adapt system to meet the response goals.

Table 2.4: Summary of Design Goals Related to Decision-Maker/User Knowledge and Experience

2.5.4 Design Goals Associated with the Decision Task Requirements

Models for Defining Design Goals to Support Cognitive Tasks

It is very difficult to generalize about tasks outside of very broad categories (e.g., planning, situation assessment, etc.). While such categories provide a general framework for discussing common error sources and failure modes, the details of HCI design remain tied to the specifics of the actual task to be supported. Carroll and Campbell (1988) suggest the psychology of human-computer interaction design is the analysis of tasks and the invention and evaluation of HCI "artifacts" to support task performance. Similarly, Woods and Roth (1988) describe cognitive systems engineering as "problem-driven and tool-constrained". In this systems-oriented view, the requirements analysis process describes the cognitive tasks and performance context then attempts to trace the causal factors associated with both satisfactory and unsatisfactory performance. The goal of this process is to raise issues for addressing these causal factors in both the system and the HCI design. While there remains no adequate theoretical basis for a prescriptive approach to design, the empirical literature provides some insights into broad categories of causal factors (e.g., attention, memory, and workload). Unfortunately, the design responses to these factors are highly task- and context-dependent and, thus, often do not generalize to other tasks or contexts.

Norman (1983) defines the means by which designers and users understand and interact with computer-based systems in terms of the construction and use of multiple models. The designer develops a conceptual model of the target system that accurately, consistently, and completely represents that system. Although Norman does not discuss requirements, the designer's conceptual model is built upon a mental model of the domain and task requirements that is often imprecise, inconsistent, and incomplete. The interface design presents a system image intended to convey information about system operation. The user constructs a mental model of the system based upon interaction with the system and system image as represented in the interface. The user's mental model of the system may not match the designer's conceptual model. Note also that the user's mental model of the task domain is determined by training and experience and, thus, varies in accuracy, consistency, and completeness.

While Norman's terminology tends to be somewhat confusing, it highlights the points at which the translation of requirements to design breaks down. The completeness of the designer's conceptual model of the system is a function of his/her understanding of the system architecture, not the requirements of the task domain. Thus, the users' mental models of the system, constructed through interaction with the system and the interface (system image), may or may not compliment their model of the task domain. The "transparency" of the interaction, that is the degree to which the users perceive themselves to be interacting directly with their tasks, is determined by the convergence of these various models. Mismatches in HCI design reduce transparency such that the user is more occupied with the operation of the system than the performance of the task (Wright & Monk, 1989).

One of the fundamental strengths of human decision-makers is their ability to conceptualize or construct mental models of causal relationships. This ability lies at the heart of intuitive and analogical reasoning. The concept of a "mental model" appears with various definitions, taxonomic structures and applications in the cognitive process literature of the early 1980's. Johnson-Laird (1983) discusses "mental models" as analogical representations for deductive inferencing tasks. One of the principle contributions of this work was to emphasize the semantic aspects of thought. Gentner and Gentner (1983) propose a "structure-mapping" theory to explain the cognitive processing of analogies. Their research employed protocol analysis and experimental manipulations to demonstrate the difference in domain understanding resulting from differing causal explanations of physical phenomena. Carroll and Olson (1988) review the mental model literature and offer a practical definition of mental models. In their definition, a mental model

- incorporates "a rich and elaborate structure;"
- involves an "understanding of what the system contains, how it works, and why it works that way;" and
- provides a way "try out actions mentally before choosing one to execute." (Carroll & Olson, 1988, p. 51)

The concept of "running" a mental model is roughly analogous to the mental simulation activity described in Klein's (1993) Recognition Primed Decision (RPD) model.

When used as an analog, a mental model serves as an 'advance organizer' for the interpretation of novel concepts (Mayer, 1979; Mayer & Bromage, 1980). Anderson (1983) suggests that analogy and the creation of mental associations may be the only way that people learn. Bott (1979) found that users will generate their own analogies to explain explain system behavior if none are provided. Research indicates that an inaccurate mapping between the user's model and the actual functioning of the system can increase task complexity and result in performance errors (Carroll *et al*, 1988). Lehner and Zirk's (1987) experimental studies involving expert system users found that an accurate mental model of system processes was key to cooperative problem-solving performance. Moreover, the high performance attained with an accurate mental model continued even when the user's problem-solving method was different than the expert system's.

Decision models such as Klein's (1993) Recognition-Primed Decision (RPD) model and Rasmussen's (1986) Skills-Rules-Knowledge (SRK) model propose conceptualization and analogical reasoning as the means by which decision-makers respond to novel situations. Similarly, conceptualization is a common factor in all four phases of the SHOR decision paradigm. The analog selected serves to reduce cognitive demand by identifying and structuring the relevant information and filtering out the irrelevant information. The mental models associated with the proposed analog then provide the means for mentally simulating the potential outcomes of the available options. Although this model appears to explain much of what makes for expert decision performance, there are several potential pitfalls. For example, the selection of an analogy may be affected by its availability in memory due to its vividness or recent experience. The selection of and adherence to a incorrect analogy may blind the decision-maker to relevant information (i.e., information that contradicts the working hypothesis). Subsequent mental simulations built upon these incorrect assumptions could mislead decision-makers with respect to the potential effects of their actions. Finally, the ability to "run" complex mental models is constrained by the limitations in human working memory and information processing capability. In highly complex domains with extensive interactions among the various factors, the mental simulation required may be intractable.

The cognitive science literature presents numerous descriptive theories and empirical studies that attest to the existence of mental models (c.f. review in Staggers & Norcio, 1993); however, there remains no systematic method for satisfactorily harnessing the power of mental models to guide the design of HCI for decision support. Several difficulties in the practical application or manipulation of mental models negatively impact their prescriptive value in HCI design (Leiser, 1992; Norman, 1983). In practice, mental models are fragmentary and lack discrete boundaries or formalized definitions. The incomplete and disconnected aspects of a mental model permits the incorporation of contradictory, non-rational, and invalid concepts. Furthermore, mental models of rarely used systems or procedures can deteriorate over time due to forgetting.

In complex, dynamic environments, the interaction models required for human-computer cooperative decision-making must assist the decision-maker in maintaining situational awareness and understanding the short- and long-term consequences of decisions. This implies a framework of models in the mind of the user that must be represented in the interaction and interface design. These include:

- **task interaction models** - representation of the current state of the target domain (situational awareness), means for acting on the domain (task variables), and means for predicting the consequences of actions on the domain (outcome simulation); and

- **system interaction models** - representation of the current state of the system and the means to understand the actions required to perform tasks using the system.

Carroll *et al* (1988) proposes a structured methodology for designing effective interface metaphors that provides a useful starting point for developing interaction models. Extending this method to the design of HCI for decision aiding suggests the following basic activities:

- Identify potential task domain models - e.g., network models for route planning;
- Describe the match between models and the domain in terms of user task scenarios - i.e., the constraints and affordances implied by the analogy;
- Identify the potential mismatches and their implications - i.e., where are the gaps or breakdowns in the analogy; and
- Determine the appropriate design strategies to help users manage unavoidable mismatches.

The task profile (Section 2.3.6) characterizes functional tasks along four dimensions (input, output, response, and feedback) and decision tasks in terms of four decision phases in the SHOR paradigm (stimulus, hypothesis, option, and response). The summary Tables B-10 through B-12 in Appendix B present definitions and evaluation scales for the multiple components of each dimension. Although the tables also suggest potential HCI impacts of these dimensions, coherent design for information presentation and interaction cannot be derived from the assemblage of individual "fixes." Woods and Roth (1988) refer to this as the "prosthesis approach" to design. In contrast, they suggest that the design goals of cognitive systems engineering focus on extending the human decision-maker's conceptual abilities.[7] Towards this end, the tables serve to identify the cognitive areas that the individual characteristic may impact (e.g., attention, situational awareness, etc.) and make some suggestions regarding the HCI design features that address those areas.

A full explication of the possible HCI design responses suggested in these tables is beyond the scope of this work. Instead, this section focuses on the identification of high-level design goals in the support of cognitive task performance and the information presentation and interaction solutions suggested by the empirical and experimental literatures in HCI design, decision aiding, and cognitive psychology.

High-Level Design Goals for Decision Task Requirements

The human decision-maker's ability to meet the cognitive demands of decision tasks is determined by both the quality of their conceptual skills and the cognitive resources (e.g., attention and memory) they can bring to the task. HCI design goals for supporting decision tasks fall into two general categories: those related to enhancing human decision-makers' understanding and those related to reducing the negative effects of human cognitive limits. These categories are actually two sides of the same coin, rather than distinctly different constructs. The first category involves what Woods and Roth (1988) term "conceptual tools." In the context of HCI design, conceptual tools are those features of the design that enhance the decision-maker's ability to structure the problem, formulate the goal, select a solution path, and implement the selected response. The second category addresses the limits of human attentional and memory resources that interfere with effective use of the human cognitive strengths that aid conceptualization. The decision-maker's attempts to cope with his/her own

---

[7] This concept is consistent with Zachary's (1988) approach the design of the knowledge representation, data management, and analytical methods for decision support systems.

cognitive limits often result in erroneous problem formulation and option selection. On the other side, enhancing conceptualization reduces certain aspects of cognitive demand and, thus, reduces the cognitive resources required. The HCI design goals proposed here also play such dual roles.

One of the most effective means of enhancing decision-maker's conceptualization is to structure the problem representation to highlight the values and relationships between the relevant task variables. Wood and Roth (1988) suggest that the extent to which designers can successfully structure representation is a function of three factors:

- the designer's ability to anticipate the decision tasks and situational variables;
- the characteristics of the representation that influence decision performance; and
- the degree of domain variation in the relationship between key criteria and decisions.

Task analysis helps to identify the decision variables and map their relationship in the decision process. Representation impacts the decision-maker's ability to monitor, perceive, combine and relate data to the assess the situation and formulate an appropriate response. In this way, structure of the representation makes the semantics of the domain visible (e.g., the ecological interface designs proposed by Vicente and Rasmussen, 1992). The third factor addresses the issue of representational economy and variety. In complex, dynamic environments the decision-makers' requirements for problem views may change given the situational context (e.g., routine operations versus crisis). In addition, representational structures may have to provide multiple perspectives on the problem.

The use of representation to aid conceptualization lies at the heart of several approaches to structuring decision variables. Treu (1992) presents examples of several structural primitives and composite structures. Each primitive is considered with respect to its effects on cognition and memory and its representation in computer-base systems. For example, node and arc structures may imply paths, scripts, spatial location, or distance. When combined with the vertical hierarchy primitive (suggesting concepts of rank, ordering, levels of abstraction, etc.) the composite structure conveys a tree or object hierarchy.

Configural or integrative displays combine the low-level (syntactic) data to form a high-level (semantic) representation. The goal of integrative displays is to facilitate the decision-maker's wholistic perception of domain or situational features that are not apparent when the data elements are separated. Initially, the concept of configural displays focused on the benefits of data proximity in object displays (Carswell & Wickens, 1987). Recent research indicates that it is the "emergent features" of the configural display, rather than the mapping to a recognizable object, that determines the benefits in performance (Sanderson et al, 1989). The value of integrative displays has been questioned where decision-makers must also attend to individual data elements (Bennett and Flach, 1992). Bennett et al (1993) empirically demonstrated the benefits of configural displays to promote extraction of both high-level and low-level data.

In their Ecological Interface Design (EID) method, Vicente and Rasmussen (1992) also incorporate integrative displays derived from an abstraction hierarchy of the work domain. Based on improvements in decision performance in an experimental study, they suggest that the representation based on the abstraction hierarchy provides a better match to the decision-maker's mental model of the work domain. These findings support the benefits of multiple problem perspectives for decision-making. Support for the multiple perspective design also applies to the research in integrative displays. Coury and Boulette (1992) investigated the effects of configural displays on diagnostic tasks in conditions involving time pressure and uncertainty. Their findings suggest that accurate and timely situation assessment under all conditions of time stress and uncertainty requires both integrated and separated displays.

In cases where representation requires multiple screens, Woods (1984) proposes that the integrative construct is the level of "visual momentum" the information presentation and interaction supports. When visual momentum is low, information processing occurs in a series of unintegrated data views requiring the decision-maker to re-orient and search for relevant information in each view. This adds to the decision-maker's cognitive workload and degrades performance. Woods suggests several structural features to increase visual momentum, including the "long shot" or overview display, perceptual landmarks, display overlap, spatial organization, and spatial cognition. Overviews, landmarks, and overlap provide information about the location of one display with respect to another and support the multiple perspective aspects of the Rasmussen (1986) abstraction hierarchy. Spatial organization uses spatial orientation (e.g., hierarchies, paths, and maps) to serve as pre-organizers and aids for exploring the domain or situation. Spatial cognition refers to the use of analogical representations to provide a map to the all the features of the underlying process. Woods suggests that increasing visual momentum reduces mental workload, improves data sampling behavior and identification of relevant information, and improves the cooperation between the human decision-maker and the computer-based support.

Representation structure also affects the cognitive demand associated with decision tasks. Norman *et al* (1986) present strategies for cognitive layouts in windowing designs that address selective attention, multi-cue integration, and variable levels of cognitive control. The layout of windows to reduce the demands of monitoring multiple activities is based on a model of attention that suggests that attention works as a dynamic filter. When multiple signals are present, attention is focused on one signal while the remainder of the signals are attenuated. The shift of attention may be voluntary (e.g., when the decision-maker actively searches for necessary information) or involuntary (e.g., flashing alarms, etc.). Spatial layouts for multi-cue integration may achieve all or part of the integration (e.g., integrative displays) or leave the integration to the decision-maker. Although integrative displays are quite powerful, effectiveness depends upon the extent of domain-criteria variability. Leaving integration to the decision-maker is the most flexible approach for the designer, but places the burden of integration entirely on the decision-maker. Finally, the use of spatial layouts may be used to provide multiple perspectives of the problem or domain from the syntactic to the semantic (i.e., the signals, signs, and symbols).

Another approach to representation is a decision-oriented or goal-oriented display. In decision-based representation, displays present problem information structured to aid interpretation. Similar to the concepts in integrative displays, these display paradigms shift much of the cognitive demand in data integration and interpretation from the human decision-maker to the computer. In contrast to data-oriented displays that present all available information, decision-oriented displays provide only the information that is relevant for the decision task. In an experimental study involving multi-phase decisions in a complex, dynamic environment, MacMillan and Entin (1991) found that decision-oriented displays resulted in faster decisions with fewer errors. Goal-oriented displays represent the domain or situational structures that relate to desired goal or system state. These can take the form of goal and sub-goal hierarchies or diagrammatic views of the system or process. Kieras (1992) employed diagrammatic displays for diagnostic tasks in control system management. Experimental investigations indicated that the causal structures and one-to-one mapping of component state to the diagram produced better diagnostic performance than the more traditional representation in which the diagram and component state values were separated.

Goal-oriented representations may also be used to support the mental simulation required to identify causes for situation assessment, evaluate consequences of options and plan for response coordination and implementation. Woods and Hollnagel (1987) present a methodology for constructing goal-means networks that incorporate the task goals, functions (the means to achieve goals), and requirements that instantiate new goals based on what the function needs to accomplish the higher goal. Woods and Roth (1988) propose goal-oriented displays for evidence processing, situation assessment, and planning. Bainbridge (1988) discusses problem representation as structure-function and goal-means networks. These graphic representations use hierarchies and cause-effect links to support pattern recognition, planning and prediction, and semantic organizers.

Mental simulation is important not only for time-pressured situations, but also where feedback is delayed due to the inherent response latency of the system. Hoc (1989) describes the problems unique to long response latencies. In such environments, the diagnosis and response to changes in a system cannot be effected in direct cause and effect relationships. The decision-maker cannot directly manipulate the goal variable, but must manipulate it indirectly through causally-related variables. Planning is complicated by uncontrolled and unanticipated interventions in the causal network and long delays in response effect and feedback. The mental simulation required for planning in this context rapidly becomes intractable without aiding.

Ball and Ord (1983) present a graphic planning aid to support the mental simulation required to predict the consequences of options in an air traffic control task. Their aid presented two problem views: the current situation with radar and a predictive display of the planned response. Bell and Ord emphasize the problem of dealing with the multiple realities of the present and predicted situational displays. Their planning aid handled these representations as discrete displays and featured both manual and computer-generated options. Experimental studies with air traffic control teams revealed decision-making problems associated with requiring the controllers to relate information from both the planning and the situation displays. This design forces the decision-maker to choose between maintaining situational awareness and evaluating the consequences of his/her response.

In a cooperative decision system, the design of the cooperation in the allocation of tasks between the human decision-maker and the computer-based support is a key factor in reducing the decision-maker's cognitive task workload. This is most often accomplished by automating the attention-intensive monitoring tasks, rapid, memory- or computation-intensive tasks, or time-constrained response tasks. Task allocation also attempts to assign to the human decision-maker those tasks (e.g., inferencing and judgment) that involve adaptive, intuitive cognitive abilities. For example, in Ball and Ord's (1983) air traffic control aid, the human controller and the computer shared responsibilities for monitoring and planning. The activities within those responsibilities were allocated based on the different strengths of the human controller and the computer. The human decision-maker's tasks involved pattern recognition and maintaining situational awareness; the computer was assigned responsibility for continuous updating of the situational data and detailed trend analysis.

Most cooperative decision-making task allocation strategies involve some form of static allocation where some or all of the tasks are directly assigned to either the human decision-maker or the computer support. For example, Ball and Ord's air traffic control system featured static allocation. Recent research in cooperative decision-making also features dynamic task allocation (c.f., Andriole & Ehrhart, 1990; Klinger *et al*, 1993; and Vanderhaegen *et al*, 1994). Vanderhaegen et al (1994) present a design for human-computer cooperative decision-making that involves a dynamic activity regulation strategy based on a model of "horizontal cooperation." The concept of horizontal cooperation attempts to avoid the negative performance effects encountered with the passive decision-maker in vertical (or master-slave) cooperation (Roth *et al,* 1988). Horizontal cooperation places both human and computer on the same hierarchical level and allows explicit and implicit dynamic task allocation in much the same fashion as the human-human cooperation in team decision-making. One particularly intriguing feature of this design is the dynamic task demand estimation capability modeled on workload and performance assessment. Rather than attempting the more subjective task of estimating mental workload, the task demand estimator employs a weighted additive model of the functional task decomposition. Weights were determined empirically by expert controllers. This task demand modelling concept would appear to also have utility in the determination of task loading during the design phase.

This section presented several models for interpreting the task-related aspects of HCI design. The high-level decision task goals proposed provide

- a starting point for integrating the situational, organizational, and decision-maker/user goals and
- sign posts to the determination of the more detailed design goals associated with the specific tasks.

Table 2.5 summarizes the design goals that support the decision task requirements. The next section discusses implementation of the current HCI design concept in prototype form for evaluation and iterative modification.

The high-level design goals raise for HCI design consideration the effects of situational context, organizational context, decision-maker/users' knowledge and experience, on the cognitive requirements of the tasks the cooperative system must perform. Each high-level goal maps to a deeper layer of task- and situation-dependent design goals. In essence, these goals provide a checklist for design, implementation, and evaluation. The HCI design concept is a configuration of information presentation and interaction strategies that represent the designer's resolution of these high-level and specific design goals.

Design Goal Summary

- Structure problem representation to enhance the decision-maker's perception and understanding of the values and relationships between the key task variables.
- Provide multi-perspective conceptualization aids that make abstract or non-visible concepts and relationships visible.
- Provide decision- or goal-oriented perspectives for organizing and prioritizing tasks.
- Support mental simulation with representations of the network of problem dependencies for situational assessment, option evaluation, and response coordination.
- Allocate tasks between the decision-maker and the computer to reduce cognitive workload and support the human decision-maker's adaptive, intuitive cognitive abilities.

Table 2.5: Summary of Design Goals Related to Decision Task Requirements

2.6 Phase Five: Implementing the Interface Concept in an Interactive Prototype

2.6.1 Prototyping Design Concepts

An HCI prototype is a physical manifestation of the configuration of information presentation and interaction methods and technologies proposed in the HCI design concept hypothesized to meet the design goals and requirements identified (Figure 2.15). Developing prototypes during the early phases of system development provides a low-risk means for evaluating both the HCI design goals and implementation hypotheses. At each stage in the system development effort, the represented HCI design can be reviewed against the current version of requirements. In this way, sponsors and operational users can respond to the prototyped design to refine the requirements base and assess the utility and usability of the proposed interface for the decision tasks. Prototypes vary widely in scope and definition, from preliminary paper storyboards to functional interfaces to data. The choice of prototype depends upon the design questions that must be answered at the current phase of system development. For example, early in the development a prototype may be no more than a set of sample screens sketched on paper or a cardboard mock-up of a control panel. More commonly, the term "prototype" is applied to early functioning versions of software and hardware.

Assessing the appropriateness and effectiveness of the proposed HCI design to support the complex interactions among humans, equipment, and information within the organization often requires some form of interactive prototype. Using an interactive prototype also provides useful insight for the overall development effort. The HCI design embodies most of the system concept that is "available" to the user to guide his/her mental model of the system. For example, the HCI design incorporates such critical system design factors as:

- the representation of information regarding the situational elements external to the system (support systems, environment, threats, etc.);
- the representation of system states and feedback to the operator on results of actions taken;
- the allocation of tasks between the human operator/decisionmaker and the computer as determined by the dynamics of the situation and the requirements of the analytical methods selected to support decision processes; and
- the modes in which users may interact with all of this information to explore situations, develop hypotheses, generate options, select among alternatives, and implement their decisions.

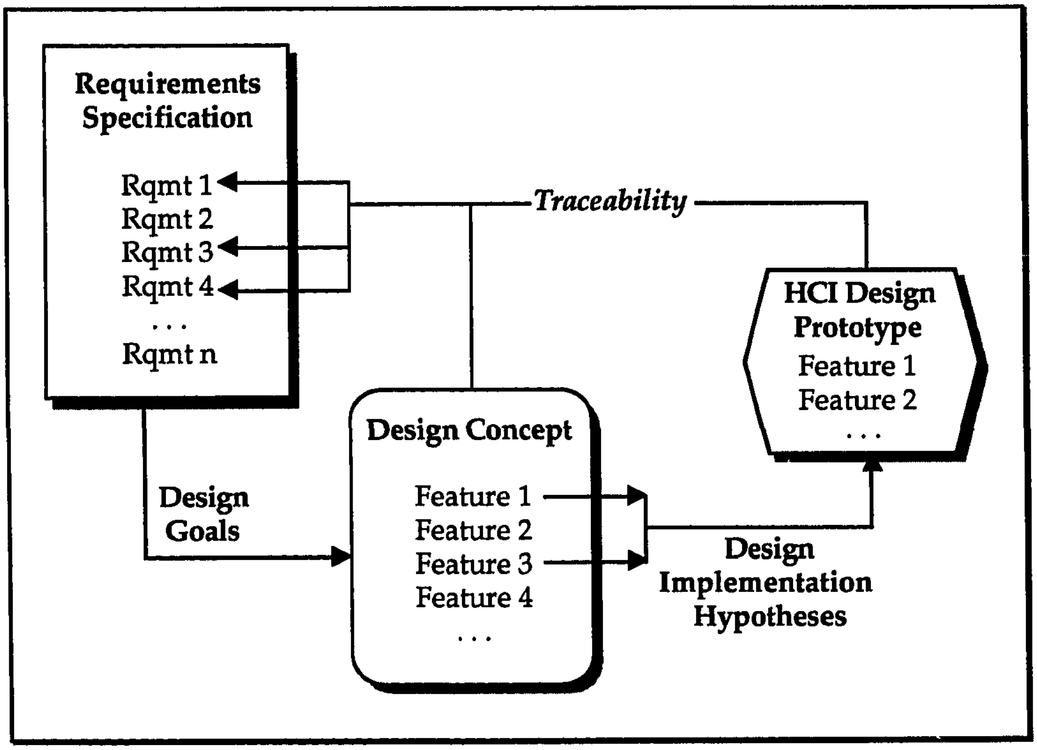


Figure 2.15: Representing the HCI Design Concept in a Prototype

In a requirements-driven design process, the judgments and decisions made during each phase determine the objectives of the analyses and evaluations required to support those decisions. Table 2.6 presents the relationship of prototyping objectives and the associated scope and boundaries of the prototyping effort. During each phase, the HCI design is considered in the context of the organizational and environmental factors which impact performance; however, these factors are represented at varying levels of detail depending upon the phase requirements. For example, during the problem definition phase and early in the requirements identification, the HCI design in question is modeled at a relatively high level of abstraction. The desired performance is expressed primarilyin qualitative terms; the nature of the interaction with other support systems and the external environment is modeled in very low detail. As development proceeds to later phases, the specification of requirements increases in detail with respect to the system itself and its interaction with other systems in the organization and with external environment. This specification, in turn, dictates the inclusion of more precise quantitative and qualitative analysis to assure that the HCI design meets both engineering specifications and organizational requirements.

HCI Design Phase	Prototyping Objectives (to support or influence)	Prototype Characteristics
Problem Definition and Requirements Identification	• Determining desirable system and HCI characteristics • Determining existing system capabilities and deficiencies • Selecting "best" of alternative system definition	• System represented at high level of abstraction • Qualitative analysis • Organizational, environmental interactions represented in minimal detail
Requirements Specification and Design	• Developing HCI requirements specifications and design alternatives • Determining "best" design	• System represented in moderate to high detail • Qualitative and quantitative analyses • Organizational and environmental interactions modeled in moderate detail
Implementation	• Determining whether developmental HCI prototype meets specifications • Providing feedback on detailed design	• System modeled in moderate to high detail • Qualitative and quantitative analyses • Organizational and environmental interactions modeled in moderate to high detail
Testing and Evaluation	• Determining whether the proposed HCI design as prototyped meets system and organizational requirements	• Qualitative and quantitative analyses • High detail in system and context modeling

Table 2.6: Prototyping Goals for System Development Life Cycle Phases

2.6.2 Prototype Implementation Strategies

The software engineering and information systems development literatures suggest a wide variety of approaches to prototyping (c.f., Andriole, 1990; Arthur, 1992; Connell & Shafter, 1989; and Nielsen, 1989). The selection of prototype form should be based on the goals of the current development phase and the information that must be derived from the prototype. Nielsen (1993) identifies the tradeoffs in prototype implementation in terms of depth of functionality (vertical prototyping) versus breadth of features (horizontal prototyping). Vertical prototyping is used in "functional" prototypes that permit the user to interact with real information; however, only a narrow range of system features is represented. In contrast, horizontal prototyping permits the presentation of the full range of system features, but without the functional capability to interact with real data.

Another common prototype classification involves the extensibility of the prototype. "Throwaway" prototypes, such as paper storyboards and mock-ups, are used in early definition phases often before the target hardware and software have been identified. The name conveys a perjorative image of sunk costs; however, the throwaway prototype facilitates communication between development teams, HCI designers, sponsors, and end-users. The information gathered not only contributes to design, but can also be used to develop instruments for the evaluation phases. "Evolutionary" prototypes involve incremental development that attempts to represent the breadth of the system with functional depth evolving incrementally. The term rapid prototyping is generally used to refer to an evolutionary prototype. Interactive storyboards are commonly used as throwaway prototypes. In situations where COTS programs and CASE tools may be used for development, interactive storyboards become the early forms of rapid, evolutionary prototypes. These four general approaches to prototyping are discussed in further detail below and summarized in Table 2.7.

Method	Advantages	Disadvantages
Paper Storyboards	• Low cost, low risk method for exploring requirements • Scenarios can be re-used for later evaluations of design • Storyboards and scenarios can later be incorporated into interactive storyboards	• Verbal descriptions in scenarios are not as vivid as visual representations • Paper storyboards support very limited exploration of interaction • May have less utility in identifying potential human errors
Mock-Ups	• Low cost method for verifying the physical layout of custom interaction hardware • May be useful in simulating environment for exercises where full interaction is not required	• Limited to representing surface features • Full capture of ergonomic aspects of performance requires more expensive representation (pushable buttons, turnable knobs, etc.)
Interactive Storyboards	• Useful for refining requirements and identifying potential human errors • Provides low- to medium-fidelity environment for performing usability trials • May be developed with low to moderate cost using COTS software	• Will not identify throughput or information overload problems associated with data volume • Designers must be careful to present only *feasible* design options within the given hardware/software constraints
Integrated Rapid Prototyping	• Useful (within limits) for evaluating performance with actual or simulated inputs • May help prevent premature "freezing" of design	• Moderate to high cost (some costs reduced when CASE tools provide easily modified prototypes) • Increasing fidelity is costly

Table 2.7: HCI Prototyping Techniques

Paper Storyboards

Paper storyboards provide a relatively low cost, low risk method for getting a preliminary feel for how the system would be used in terms of typical tasks and situations. Storyboards may be annotated, reordered or even re-designed during requirements definition interviews. Paper storyboards are limited to representation of a set scenario with little possibility of exploring the range of interaction possible with the given design. The technique presents the sequence of screens, but does not capture potential interaction errors or the cognitive workload associated with a particular design. These aspects are better addressed with interactive storyboards.

Mock-Ups

Mock-ups encompass a variety of non-functioning physical representations ranging from cardboard models of single control panels to full-scale control centers with turnable knobs and flippable switches. They are primarily used for studying the ergonomic impacts of equipment layout on physical task performance. In many cases, physical mock-ups are unnecessary for studying HCI design since most of the visible features of interest are incorporated in interactive storyboards or prototype systems. Where custom interaction hardware is required for user input or decision-makers must perform other physical tasks while operating the system, mock-ups assist in doing early evaluations of the potential workload associated with HCI design alternatives.

Interactive Storyboards

Interactive storyboards serve as a powerful means for exploring HCI design alternatives without incurring the expense of developing a working prototype. This is particularly advantageous when the investigation is focused on evaluating several advanced interaction technologies rather supporting the design of a specific system. Interactive storyboards are also useful for working with experts or end-users to refine requirements. Subjects interact with a computer-based storyboard simulating the actual operation of the system. Interaction may take the form of informal exploration or subjects may be presented with tasks to perform using the simulated system. In the latter case, the storyboard provides a low- to medium-fidelity environment for assessing usability and identifying potential human errors. Verbal protocol methods may be used to elicit the cognitive processes involved in the interaction.

Where storyboards are used in requirements definition and refinement, care must be taken not to present something in storyboard form which is infeasible within the technological and resource constraints of a working system. Although this method can be used to identify problems with cognitive workload due to the allocation of tasks between the operator and computer, it does not task the system sufficiently to delineate user or computer performance problems related to throughput or information overload. These issues must be addressed with operational prototypes that accept real-time data.

Rapid Prototyping

Although developing prototype versions of a system is not a new concept, until recently software prototyping tended to be restricted to semi-operational *beta* versions of systems under construction. As such, they represented a considerable investment in time and effort and major changes to the design were highly discouraged. Furthermore, it was not uncommon for a cost-conscious sponsor to stop development with the prototype. If the prototype offered most of the functionality of the completed system, the sponsor would take delivery on the prototype and cancel further development. Similarly, if the prototype indicated major problems with the design or development effort, the sponsor might consider it good management to cut his/her losses at that point. For obvious reasons, developers grew reluctant to show prototypes to their clients.

The introduction of fourth generation languages and CASE (computer-aided software engineering) tools dramatically changed the role of prototyping in system design and development. Using the toolboxes provided in COTS (commercially available off-the-shelf) software, prototypes with complete interactive displays using windows and pull down menus can now be developed very rapidly for UNIX, DOS, Macintosh and other environments. This rapid development capability and the corresponding ease with which the software may be modified or even substantially re-designed, makes it possible for designers to develop and use prototypes during the earliest phases of design. These early prototypes provide many of the features of interactive storyboards while reducing the possibility of presenting the user with an infeasible system concept. Nevertheless, until the system is tasked with the full volume of data expected in the target setting, actual system performance and its impacts on the users will not be fully apparent. This has important implications for the reliability and validity of HCI design evaluations.

2.7 Phase Six: Evaluating the Prototype

With the growth of interactive computing and its application in support of complex decision-making, HCI design prototyping has become an important tool in capturing and analyzing user requirements. Figure 2.16 presents a break out of HCI prototype evaluation benefits. In iterative design and development processes, prototype evaluation aids in verifying and validating the working design against the requirements. Each prototyping phase culminates with some form of evaluation. As with prototyping, the evaluation goals vary depending upon the current development phase. Early evaluation provides a means for extending requirements and task analyses to the evaluation of the procedures embedded in the current design solution. In this manner, evaluation provides a means for acquiring information about the current version of the HCI design with respect to the performance characteristics and capabilities of the human-computer cooperative decision system. Finally, this process of iterative design, prototype implementation, and evaluation supports the project management planning and control processes that ensure the overall development effort stays on track (with respect to the delivery of a quality product) and within the cost and schedule parameters. The cost/benefit ratio of incorporating an evaluation method depends not only upon the size and complexity of the project, but also at which point during development the prototype evaluation is conducted (Mantei & Teorey, 1988).

Feedback is a course correction device. Early evaluation allows design modification during the initial life cycle phases when the cost to modify is lower. For the design team, evaluation is also a discovery process. Findings from the evaluation provide input for requirements and design modification and help to set measures of performance (MOPs) and measures of effectiveness (MOEs), benchmark targets for later system-level evaluations. Evaluation feedback informs not only the design of the particular functions and features considered, but also provides input for the design of related components. For the project manager, evaluation feedback is a critical part of project planning and control. Early evaluation flags potential problems which may require cost, schedule or, in some cases, contract modification.

Chapter 4 presents an initial evaluation of the impacts of CSE design on both the development process and the end product. For both the designer and manager, incorporating CSE activities into the development process assures a better match to the operational need by capturing a more robust set of functional and non-functional requirements. This understanding supports informed decision-making when design tradeoffs must be made during development life cycle.

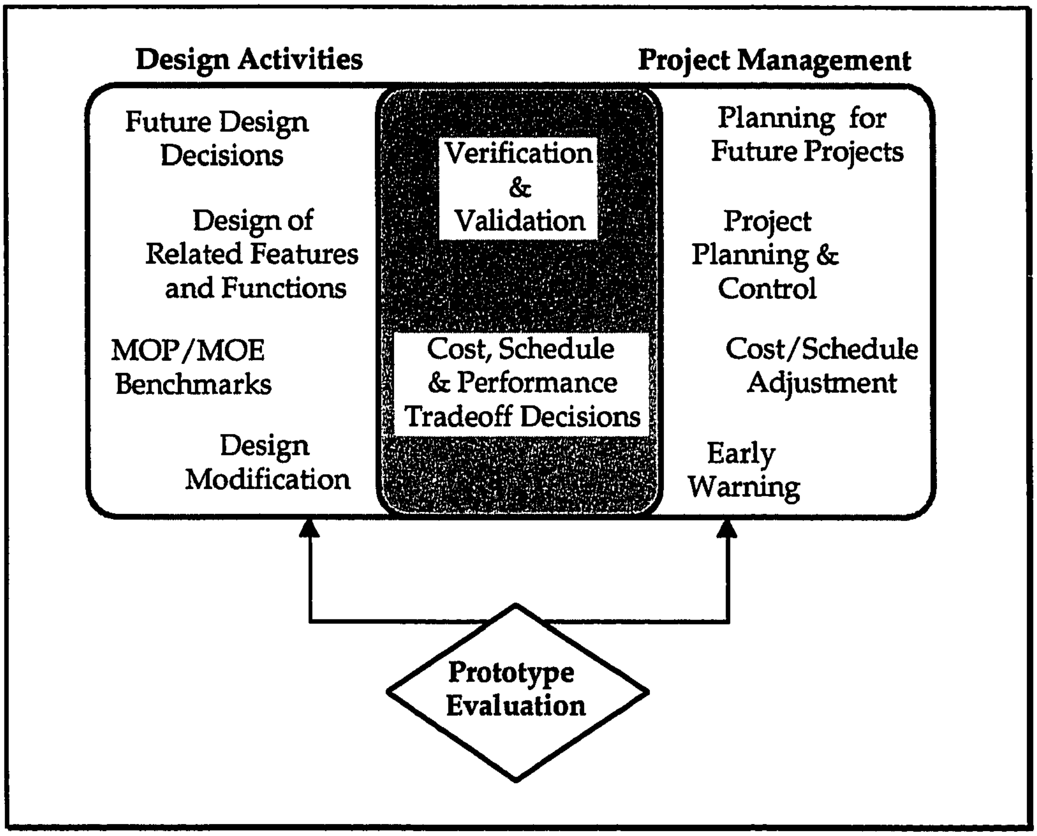


Figure 2.16: Benefits of Evaluation Feedback to Development

2.7.1 Setting Evaluation Goals

The evaluation goals of the design practitioner are distinctly different from those of cognitive science researchers. Research in the practical, real-world aspects of HCI and human-machine cooperative decision-making involve highly complex constructs. As discussed above, the evaluation of the HCI design prototype should track to the design goals as defined by the requirements (Figure 2.17). Two principal evaluations should be conducted at each level of prototyping: 1) verification of design implementation of HCI requirements, and 2) validation of design implementation's effectiveness in terms of interface usability and utility. Depending upon the design phase, the evaluation scope, and the level of detail in the design prototype, evaluation may range from designer-reviewed checklists and rating scales to empirical evaluations with representative users.

Computer-based interactive prototypes provide an opportunity for direct observation of the human-computer decision performance. Several methods are available for examining interaction processes through automated capture and analysis of interaction protocols to facilitate the rapid data analysis required for design iteration (Smith *et al*, 1993). The empirical study approach builds information in a data-intensive, bottom-up fashion. While empirical evaluations can be used to determine performance benchmarks, they do not permit direct insight into the performance *requirements*. These requirements evolve from a top-down analysis based upon the organizational and system objectives, functions, and the tasks identified with those functions. Without the analytical framework, the measures collected in empirical studies lack context and can misdirect decision-makers. In this context of these goals, Rogers (1992) questions the desirability of the micro-analysis and theoretical rigor that characterize research in cognitive psychology. Rogers suggests that applied HCI research (and, by extension, HCI design evaluation) benefits from a macro-level analysis that allows a parallel, symbiotic relationship with the theoretical aims of cognitive research.

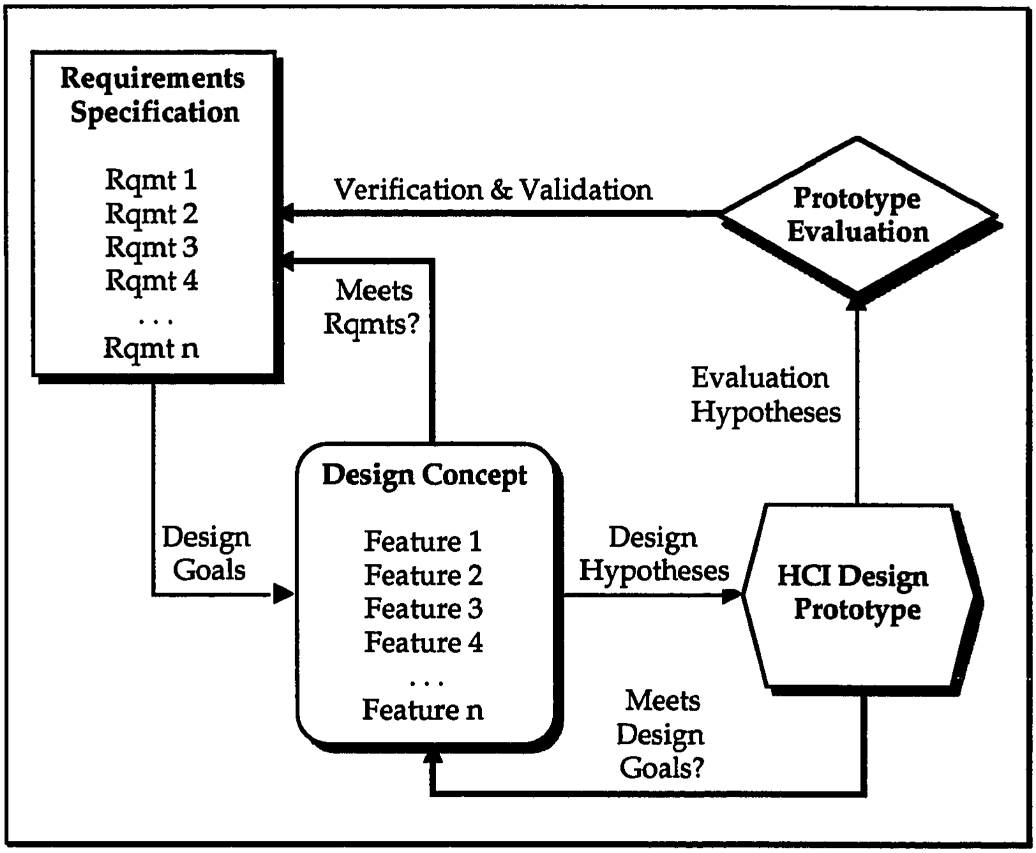


Figure 2.17: Relationship of Process Inputs to Evaluation Goals at Each Development Phase

Rasmussen and Pejtersen (1993) conceptualize the well-balanced evaluation of a cognitive system design as a combination of top-down analytical evaluation and bottom-up empirical assessments (Figure 2.18). System design evolves through the top-down analysis of the intended purpose and identified functions. Functions are then decomposed into the procedures and tasks allocated to the machine and the user, culminating in the design that maps the system's form. Bottom-up empirical evaluations first address the lower level human factors issues associated with fundamental usability and continue by evaluating the support of the cognitive requirements involved in the tasks. These human requirements interact with the system's allocation of functional requirements and the capabilities afforded by the design.

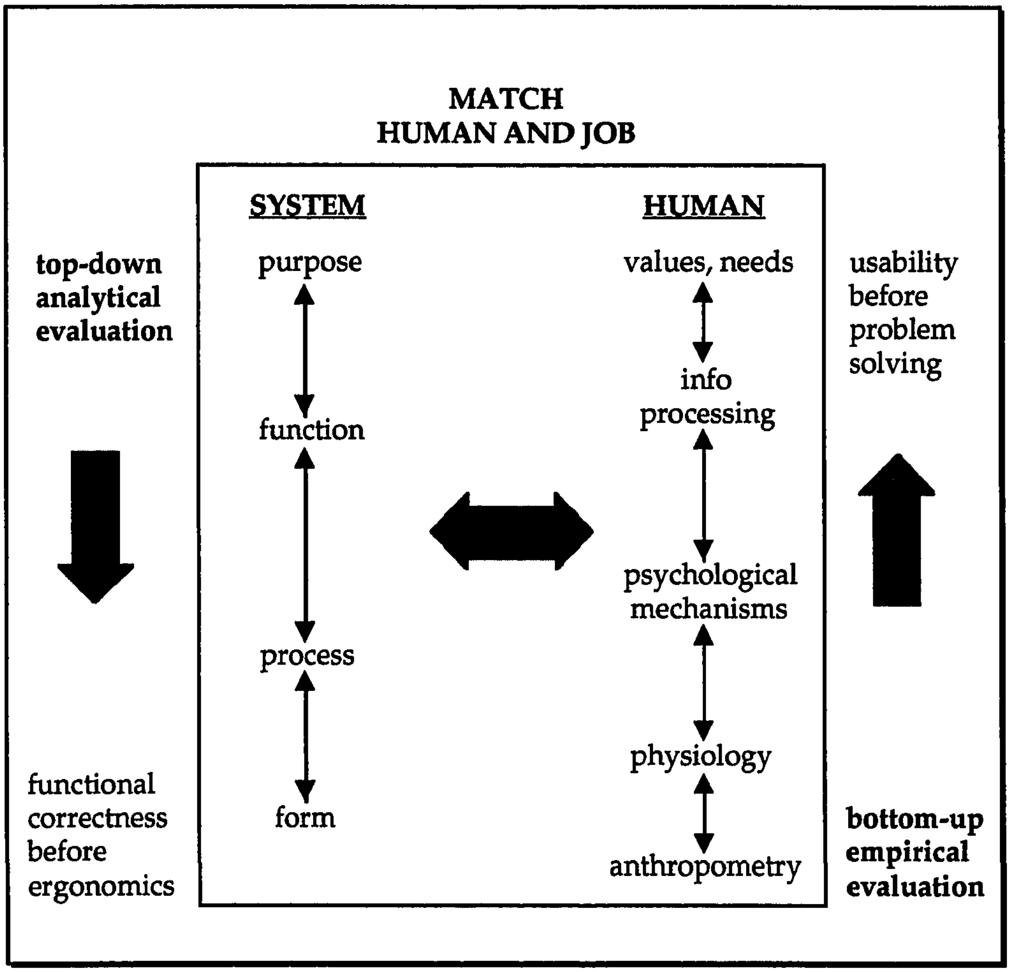


Figure 2.18: Contribution of Analytical & Empirical Evaluation Approaches
(Rasmussen & Pejtersen, 1993)

Despite some variations in terminology, this prescription for a combination of top-down analytical and bottom-up empirical evaluation is consistent with similar discussions in Meister (1985, 1991) and Adelman (1992). Meister (1985) presents a series of human performance questions grouped by development stage and indicates the various analysis and evaluation methods that supply answers. The balance between analytical and empirical evaluation approaches shifts depending upon the stage of the HCI design development. For example, in the early stages of planning and design, there is a strong reliance on top-down analysis methods supported by the available objective data and subjective judgments. The later phases of detail design and prototype testing employ more rigorous empirical evaluation methods and well-structured subjective measures to assess performance in terms of the functional requirements outlined in earlier phases of development.

2.7.2 Selecting Evaluation Methods

One of the most often used terms in HCI design evaluation is "usability" The narrow definitions limit usability to the mechanics of operating the interface. Nielsen's (1993) usability heuristics exemplify this narrow definition. In a somewhat broader definition, usability may be seen as the measure of the system design's ability to support the user in accomplishing their tasks (c.f., Ravden & Johnson, 1989; Wright & Monk, 1989). This model of usability incorporates the interface operation tasks as a subset an overall measure of the effectiveness and ease of use of the system.

Several researchers have proposed the use of so-called "discount" usability evaluation methods to identify areas for improvement early in design (c.f., Nielsen, 1993, Wright & Monk, 1989, 1991). Nielsen's (1993) ***heuristic evaluation*** essentially takes the accepted HCI design guidance (e.g., use simple and natural dialogue, provide adequate feedback, etc.) and converts it to checklists of nine usability properties. Heuristic evaluation may be performed by three to five evaluators and does not involve interaction with users. Empirical evaluations using as many as 77 evaluators indicated that aggregating the responses of as few as five evaluators resulted in the capture of 55 - 90% of usability problems (Nielsen & Molich, 1990). The research also pointed out the relatively poor performance of individual evaluators. The fundamental limitation of Nielsen's heuristics is their focus on design aspects of interface operation. As designed, the checklists do not provide the means to examine the extent to which the design addresses the cognitive task requirements.

Nielsen's (1993) text on usability engineering discusses usability heuristics and heuristic evaluation, but does not present example checklists or present sufficient information to guide the conduct of heuristic evaluation. Ravden and Johnson (1989) present a more comprehensive evaluation that employs nine usability criteria. Each criterium is addressed in ten to twelve questions that may be amended to address the specific evaluation goals. Evaluators include the designer(s), representative end-users, and other technical professionals (i.e., human factors experts, etc.). The checklists are completed individually by the members of the evaluation team as they perform a predetermined set of exemplary tasks. The principal advantage of Ravden and Johnson's method is the potential for rapid analysis and the ready conversion of the subjective data into quantitative measures for comparison. The most significant source of overhead is in the selection and development of interaction tasks. Depending upon the goals of the evaluation, the development of simple tasks or task scenarios may entail extensive preparation.

Wright & Monk (1991) avoid some of the shortfalls in heuristic evaluation while retaining its low cost and effort features. Although they acknowledge the value of careful quantitative evaluation, they suggest that qualitative evaluation provides more cost-effective guidance for the early phases of design. Their approach, intended for the design practitioner, involves designers and users in a ***cooperative evaluation*** using think aloud protocols and verbal probes. Analysis in this early phase is highly focused to capture the relevant information within cost and schedule requirements. Wright and Monk (1989) indicate that evidence in the form of either critical incidents or breakdowns is sufficent to identify HCI design problems. Not to be confused with the retrospective analysis technique used by decision researchers (i.e., Klein, 1989a), in this context a ***critical incident*** is some user behavior that fails to use the functionality of the system efficiently. A ***breakdown*** designates any point in the interaction where the user's focus on the task is broken due to the demands imposed by the system (i.e., the interface ceases to be "transparent").

The ability to use the designer as evaluator provides the speedy, inexpensive evaluation necessary for iteration in the early stages when the design is evolving rapidly. Experimental investigations performed with design trainees indicate that satisfactory rates for detecting design problems may be achieved quickly by designers with little or no human factors background and limited training in the method itself (Wright & Monk, 1991). Rather than merely endorsing their own designs, the results of the study indicated that designers were better at evaluating their own systems using this method than similarly experienced evaluators not associated with the design. Furthermore, the designer-evaluators uncovered more unanticipated problems than the evaluators not involved in the design. The principal limitations in the cooperative evaluation method include problems with the task altering aspects of think aloud protocols and the potential for bias in the single designer-evaluator model. Similar to Ravden and Johnson's usability evaluation method, cooperative evaluation also requires the preparation of meaningful tasks to provide the context for the evaluation sessions.

Departing from the design phase orientation of the classic SDLC model, Gardiner and Christie (1990) examine the role of prototypes in addressing questions on four HCI design levels: conceptual (the system concept), semantic (the interaction concept), syntactic (the interaction form), and lexical (the interaction detail). In related work, Ehrhart (1993) presents a survey of evaluation methods useful for assessing HCI designs to support human-computer cooperative decision-making. Gardiner and Christie's model provides some useful guidelines for trading off the time and expense required for developing a prototype against the functionality and performance achieved. In addition, it indicates the extent of evaluation support possible with a relatively small investment. Table 2.8 combines the suggestions of Gardiner and Christie (1989), Nielsen (1993), and Ehrhart (1993) for linking the proposed evaluation focus, prototyping support, and evaluation techniques appropriate at each design level.

This chapter presented a framework for employing cognitive systems engineering methods to define problems, identify and represent cognitive task requirements, develop design goals, and implement and evaluate HCI designs for information presentation and interaction in human-computer cooperative decision systems. The next chapter applies the CSE framework to a real-world development project involving a cooperative decision-making in a complex, dynamic environment.

Design Level	Design Evaluation Focus	Prototyping Support	Evaluation Tools & Techniques
Conceptual	• System and HCI design concept • Appropriateness for user requirements and abilities	• Written descriptions and scenarios • Storyboards • Interactive storyboards	• Focus groups • Walk-through • Predictive models • Heuristic methods
Semantic	• Interaction concept • Broad definition of interaction, error feedback and user support	• Interactive storyboards • Hardware mock-ups • Partial prototypes	• Informal user tests and observation • Walk-through • Checklists and rating scales
Syntactic	• Interaction form • Dialogue parameters and interaction sequences	• Interactive storyboards • Partial (developmental) prototypes	• Formal and informal user tests • Walk-through • Controlled laboratory tests • Field tests and observation
Lexical	• Interaction detail • Specification of HCI	• Partial (developmental) prototypes • Functional prototypes	• Formal user tests • Gaming & Simulation • Field tests and observation

Table 2.8: Prototyping and Evaluation to Match HCI Design Requirements
(Christie & Gardiner, 1990; Ehrhart, 1993; Nielsen, 1993)

3. HCI Design Application:
The FLEX Tanker Module Case Study

The development of the Force-Level Execution (FLEX) prototype at the Air Force's Rome Laboratory (RL) presented an excellent opportunity for applying and evaluating the CSE framework for HCI design. The FLEX Program is a collaborative rapid prototyping effort between the Advanced Concepts Branch in the RL/C3 Division and industry contractors. Officers from the US Air Force's major operational commands in the United States, Europe, Pacific and the Far East participated in a review board known as the FLEX Working Group (FWG) to provide an operational end-user perspective. The RL development team took responsibility for developing the user interface.

At the point this case study begins, the FLEX designers had just completed and demonstrated to the FWG the first of three functional prototypes. The prototypes allowed realistic interaction at network-linked workstations with a representative data set. This permitted the FWG to try out the system concept and provide feedback on the information presentation, task interaction, etc.

This chapter takes the reader through the development of an alternative HCI prototype using the methods described in the previous chapter. The CSE framework was employed to define the problem, identify and model the cognitive task requirements, integrate the requirements into the System/Segment Specification (SSS), translate requirements into HCI design goals, create an interactive prototype of a CSE-based version of the FLEX interface, and develop a plan to evaluate the prototype against the existing FLEX interface.

3.1 Defining the Problem

The first step in problem definition involved defining the FLEX system and placing it in context within the organization. Most of the initial definition was based upon an early version of the FLEX System/Segment Specification (SSS)[1] and the trip reports written by the RL development team after their visits to air operations centers in the United States, Europe, the Pacific and the Far East. Additional information was drawn from the demonstration of the first FLEX prototype for the FWG members.

As indicated in Figure 3.1, FLEX is part of a suite of systems that supports the Combat Operations Division (COD) of the Air Operations Center (AOC) in the planning and execution of the air missions. FLEX receives the mission plan from the Combat Plans Division in the form of an Air Tasking Order (ATO). Formatted in machine-readable text, an ATO for 24 hours of combat missions may run into hundreds of pages. While the text form permits rapid transmission to the operational units, the ATO is unwieldy and does not provide a sense of the actual mission flows (Figure 3.2). To compensate, planners and operational staff officers manually create a variety of charts and maps to display mission data in a form that permits multiple data views. The Advanced Planning System (APS) allows the planning team to work with the details of the ATO data using tabular displays of the relevant data bases and automating many of the calculation and charting operations. Since the planners and operational decision-makers use much of the same data and knowledge bases, APS and FLEX share many common screen layouts.[2] The common windows help promote consistency across these two closely-coupled systems.

---

[1] The SSS is the standard document for system-level requirements documentation under the Dept. of Defense software development standard, DOD-STD-2167a.

[2] Figures G-1 through G-7 in Appendix G present modified versions of the existing FLEX system windows that apply to the Tanker Operations tasks addressed in this case study.

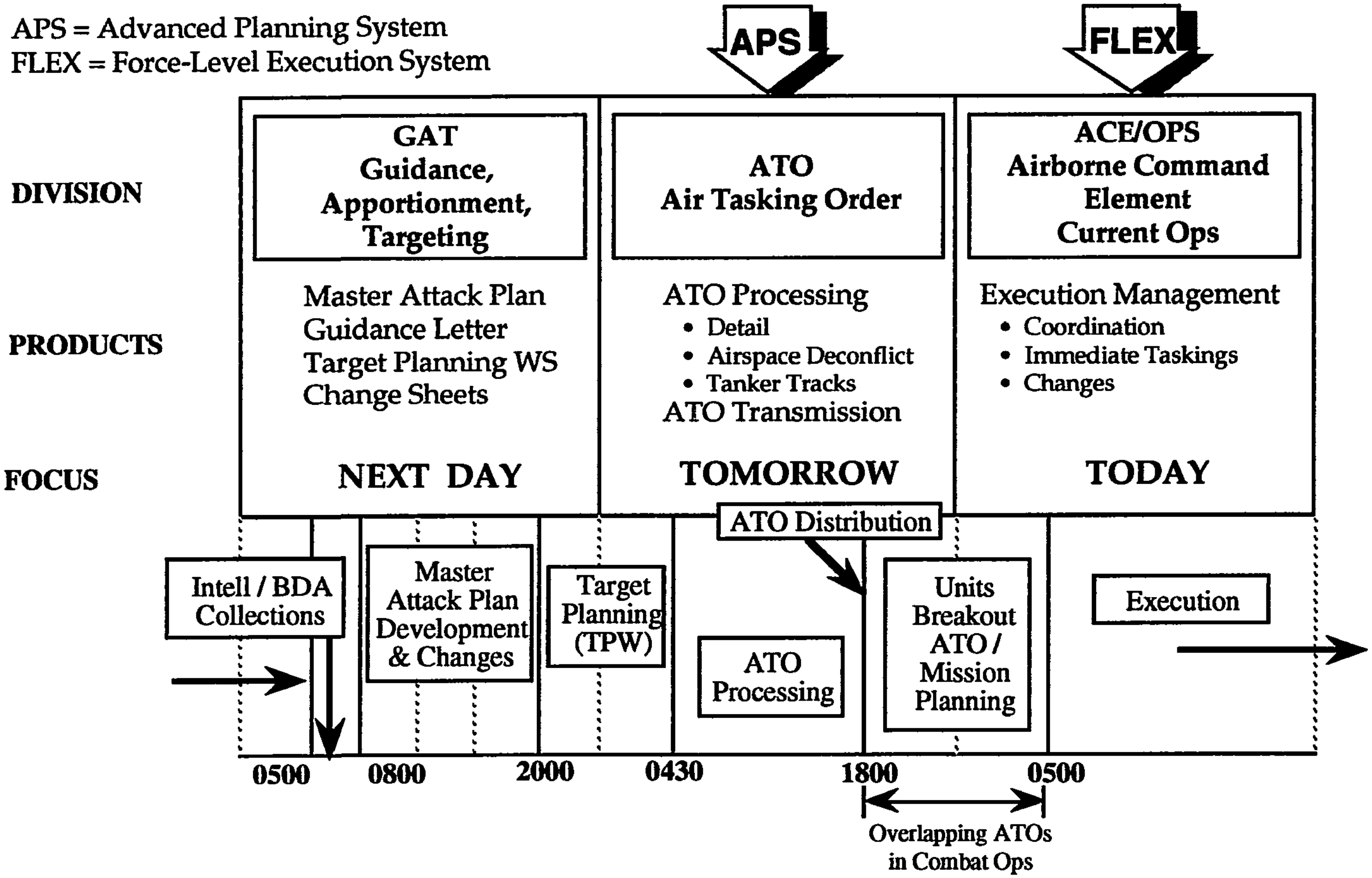


Figure 3.1: FLEX & the ATO Timeline

```
* * * *    UNCLASSIFIED    * * * *
EXER/FLEX CASE STUDY//
MSGID/ATOCONF/TACC//
PERID/150600Z/TO:160559Z//
AIRTASK/UNIT TASKING//
TASKUNIT/25ARS/OBBI//
MSNDAT/11/-/LINKAGE00/1KC135R/AR/ /BOM/ /126/23600/33600//
AMPN/SHELL//
MSNLOC/151130Z/SHELL/ALT:170/-/2811N04650E//
7REFUEL
/MSNNO    /ACSIGN       /NOTPAC          /OFF    /ARCT    /TNKR  /FUEL/CMNT
/39       /NOSEGAY  05/2A10A             /12     /        /      /JP4/PRE
/39       /NOSEGAY  05/2A10A             /25     /        /      /JP4/MID LATE
/62       /NOSEGAY  11/2A10A             /12     /        /      /JP4/PRE
/62       /NOSEGAY  11/2A10A             /12     /        /      /JP4/MID LATE
//
. . .
//
TASKUNIT/353TFS/OEPA//
MSNDAT/46/-/NOSEBAG01/1A10A/XCAS/-/B5/-/124/23301/33301//
MSNLOC/150800Z/151000Z/HANDEL/ALT:200/-/2840N05535E//
REFUEL/ROMAN00/21/ESSO/ALT:170/-/10//
REFUEL/ROMAN00/21/ESSO/ALT:170/-/11//
REFUEL/ROMAN00/21/ESSO/ALT:170/ /2//
. . .
```

Figure 3.2: Examples from an Unclassified Air Tasking Order

The combat air operations decision environment is complex and dynamic, involving a high degree of uncertainty combined with time pressure and high threat. The duty officers (DOs) in the COD monitor the execution of the ATO missions and re-plan as required to meet changes in goals and/or available resources.[3] The various air missions are so interdependent that changes in the availability of a support mission can result in the cancellation or re-scheduling of attack and support missions across the entire ATO. This "ripple effect" makes timely re-planning extremely difficult.

The first models developed for problem definition decomposed the monitoring and control, planning, and communications tasks performed by the COD

---

[3] The Duty Officer in AOC is a decision-maker, thus, in discussions of FLEX, the term decision-maker (DM) is used interchangeably with duty officer (DO).

decision-makers in terms of their representation in the first FLEX prototype and the trip reports. Figures C-1 to C-15 in Appendix C present examples of the high-level models of the COD tasks and interactions. These models were iteratively refined as the requirements were identified.

To provide a tractable example, the CSE case study focused only on the FLEX re-planning support to the Tanker Duty Officer (TDO). The TDO is responsible for providing air refueling (AR) support to all scheduled missions that require refueling. Re-planning is required when new missions are created, existing missions re-routed, or air refueling resources change. The TDO performs re-planning tasks as indicated by their own assessment of the evolving situation and as tasked by other duty officers. Figures C-16 through C-20 in Appendix C present examples of the cognitive maps developed to model TDO tasks and decision variables. Although these models were roughed out during the problem definition phase, most of the detail was developed as part of the in-depth analysis conducted during the requirements identification phase.

3.2 Identifying and Modeling the Tanker Duty Officer's Cognitive Task Requirements

The case study was external to the actual FLEX development effort; therefore, the cognitive task requirements identification process began with the examination of system requirements information gathered from a variety of sources including:

- **Document Reviews** - Rome Laboratory (RL) development team trip reports, FLEX statement of work, contract developer's system/segment specification (SSS) and system software design documents, written change requests, and a variety of Air Force manuals and support materials on air refueling operations were reviewed. (See Appendix D for a complete list of documents consulted.)

- **Interviews** - interviews were conducted with RL team, the contract development teams, FLEX working group (operational personnel from major commands), and tanker operations personnel from Griffiss AFB's 509th Air Refueling Squadron.
- **Observation** - observations were made of FLEX Working Group (FWG) officers interacting with early prototype versions of the FLEX interface.

Referencing the tables in Appendix B, the user, organization, task, and environmental/situational models evolved into a set of cognitive task requirements (CTRs) that became the design objectives for the CSE interface prototype. These materials were used to iteratively refine the models of the air refueling domain, the TDO and the tanker re-planning tasks (Appendix C).

The remainder of this section reports the requirements defined by applying the information in the Appendix B tables to the information gathered from documentation, interviews and observation.[4] A number of graphic hierarchies and taxonomic models were created as the requirements evolved. These were used to develop an understanding of the procedures and information required to accomplish the re-planning tasks.

3.2.1 Defining the FLEX Environmental/Situational Context

Using Tables B-1 through B-4, it was possible to characterize the FLEX environmental/situational context in terms of its inherent structure, determinacy, boundedness, and complexity. In combat situations, decision-makers in the COD must cope with an environment that ranges from severely stochastic (e.g., the coordination of a complex array of friendly assets) to indeterminate (e.g., mission perturbations caused by an intelligent adversary). There is a high degree of

---

[4] References to individual tables in Appendix B use the table number, including the appendix prefix (e.g., Table B-1).

variability in all the ATO plan components. For example, the decision variables are generally representative, but differ substantially in reliability due to timeliness of updates or their inherent ambiguities. Due to factors such as uncontrollable environmental conditions and the existence of intelligent adversaries, it not possible to completely control the outcomes by manipulating the initial conditions. Thus, the re-planning environment tends to be open and ranges from semi-structured to unstructured due to the high volume of information and potential for "unknown unknowns". Most of the information load under routine conditions is tractable for a well-trained and highly-motivated TDO. Under combat conditions (e.g., 2000 sortie ATO) the tasks become intractable, with information loads exceeding human ability to absorb and manipulate.

Situational complexity ranges from moderately high to very high depending upon the nature and size of the operation. Air refueling is a pervasive support activity and tanker missions are the "tent pole" in air operations. Moreover, tanker operations involve a secondary network of dependencies. The fuel a tanker has available for refueling (i.e., taskable fuel) is dependent upon actual fuel offloads that, in turn, are dependent upon the specific receiving aircraft and the nature of the missions involved. An inability to meet refueling requirements will result in cancellation of missions (direct dependency) with a ripple effect upon the missions which the canceled missions support (indirect dependencies). Due to these extended dependencies, the situational picture becomes less reliable as multiple changes to the ATO are effected during combat execution. As a result, the question is not *whether* the ATO will unravel, it is *how much, in what ways*, and *when* it will unravel.

This complex situational context has several impacts. First, in response to the domain, the organization must develop the means to make most efficient use of resources in a succession of varying short-term situations. Moreover, the decision-makers must be able to rapidly and effectively exploit opportunities and retain maximum flexibility and adaptiveness in novel situations. There is a potential for misallocation of resources due to the latency between recognition of the situation and internal readjustment. The adaptive strategies required (e.g., rapid re-tasking) may be difficult to coordinate and control due to complex missions interdependencies. Furthermore, achieving the required flexibility may negatively impact the ability to exercise control. Finally, organizational learning may be impaired by the lack of repeated experiences.

To meet the organizational response goals and potential errors, the decision-makers must be provided with information to help them understand the structure of the domain and current problem. For example system-level (i.e., tanker operations) overview displays can relate functional relationships and provide externalized mental models of the operational domain. Decision-makers also need the ability to adaptively filter information at the required abstraction level, while retaining rapid access to detailed information.

3.2.2 Profiling the FLEX Organizational/Doctrinal Context

The COD is part of a hierarchical organization which has both a vertically and horizontally complex chain of command with a moderately-high interdependency between functional units (Table B-4). The vertical complexity shifts to very high in joint and combined operations that require extensive coordination. The control structures in adaptive decision-making organizations shift in response to changes in the decision requirements (Table B-6b). Thus, the general tendency toward the more formal organization evidenced during routine operations shifts during crisis situations to accommodate the requirement for a more flexible response. Table B-6a presents the situational context which triggers shifts in organizational response, the effects of those shifts on decision-making activities, and the design implications for supporting this adaptive environment.

During routine operations, the situational context is determinant to moderately stochastic. The threat is low and the environment is relatively static with longer decision horizons. As a result, operations tend to be tightly controlled and decision-making is more formal. Responses to re-planning situations follow more rigid procedures based on specific guidance; therefore, the TDO is less likely to exercise a high degree of personal initiative. Control is communication-dependent and the communication delays between levels of the hierarchy lengthen the time between decision and action. Routine operations afford little opportunity to develop a range of adaptive responses as the TDO never has to push the system to the limit. As a result, during non-crisis operations, the TDO may be ill-prepared for a sudden shift in the environment to a combat state.

In contrast, during crisis operations the situational context is severely stochastic to indeterminate. The adversarial threat of destruction and mission failure is high and decision time is greatly constrained. To facilitate rapid, adaptive responses, operational control is loosened such that the informal problem-solving structures within the COD may dominate the formal structures. As the COD workload increases, the TDO will exercise more individual initiative. Although this provides the TDO an opportunity to extend his/her repertoire of response options, the subsequent relaxation of control may result in local satisficing (that is, solving the sub-unit problem at the cost of larger goals). Intra-COD communication greatly increases and the central role of tanker operations results in a barrage of task alerts to the TDO. Communication delays may impair information gathering and decision implementation required for more adaptive responses.

3.2.3 Profiling the Tanker Duty Officer (TDO)

The profile of the Tanker Duty Officer (TDO) incorporates not only their knowledge of the specific functional tasks assigned to them and their ability to operate the system, but also their understanding of goals and characteristics of the larger domain in which those tasks are performed. Table B-7 presents the defining characteristics, potential errors and system design implications associated with the user's expected level of domain knowledge. The TDO is typically an Air Force major or lieutenant colonel with a moderately high knowledge of the air operations domain acquired through experience, training, and service schools.

Many of the errors in situation assessment may be traced to the DM's knowledge of the operational context. The TDO will have situational models of the domain mostly gained through instruction and exercises and should recognize most prototypical situations. In many cases, for example, the TDO may have wing-level, but not force-level mental models. TDOs without operational experience at the wing or force level will not generally possess wholistic domain models. In addition, although domain-knowledgeable TDOs may exhibit the ability to intuitively interpret novel situations, they may not be consistent in their combination of situational cues. TDOs will generally structure goals based upon learned procedures, direct guidance, and situational models of domain and task. The extent of his/her domain understanding may limit the TDO's ability to resolve conflicts between situational models. Situations triggering multiple models may be interpreted based on the model that is more available or vivid in memory. Finally, the TDO may fail to recognize the degree of uncertainty in current information or the impacts of aggregated uncertainties on the viability of the plan.

The TDOs' knowledge of the specific functional tasks assigned them in the COD may also vary depending upon their previous experiences in combat operations (force and wing level) and training (schools and exercises). TDOs will typically exhibit high ability to perform routine procedures and moderate to moderately-high adaptability under increased workload and novel situations. Their moderate to high task experience potentially triggers errors associated with the heuristics used to reduce the high workloads during ATO execution (Table B-8). For example, in high information volume situations, moderately knowledgeable TDOs may not have adequate schema to distinguish relevant versus irrelevant information. They may also erroneously focus on task features that match stored (especially readily available) schema. Fixation on task features that match well-known (or vividly remembered) situations may prevent the TDO from correctly diagnosing the situation. Furthermore, misdiagnosis may result in the misapplication of a learned response. More experienced TDOs are still vulnerable to a general insensitivity to the potential aggregation of error in the microdecisions performed in multi-stage decision-making. For example, they may tend toward overconfidence in their current decisions and fail to revise their assessments and decisions when the situation changes. Finally, there is a general tendency for the TDO to think in serial, linear sequences rather than parallel networks of contributing causes and branching consequences of actions that make up the current situation and affect the success of the plan.

The TDOs' system interaction/operation knowledge will typically be the most variable dimension. In the absence of a protracted war, the majority of the officers assigned to the COD will be casual to competent system users (Table B-9). That is, they will not routinely have to operate the system under the time-critical, high workload conditions which characterize combat operations. Adequate operation of the system during routine or training operations will deteriorate under stress resulting in a variety of errors and an increased level of frustration and confusion. Casual system users tend to forget training without use and make mistakes (errors due to wrong intentions) and slips (errors due to unintentional actions). Casual system users rarely remember the system shortcuts that speed up performance of learned procedures and the increased workload will result in greatly impaired performance for all but simplest tasks. The competent user will be able to adapt well-understood processes to increased workload, but still have difficulty with the increase in novel situations. More competent users make mistakes by misapplying learned procedures.

TDOs with less system experience may be confused by their system operation errors. For example, TDOs may make modal errors due to a misunderstanding about current system state. A modal error involves the incorrect use of an interaction procedure that would be correct in another system state. In addition, users may "get lost" in the system, finding themselves in unfamiliar windows or locked out while the system performs an unintended procedure.

3.2.4 Profiling the TDO's Functional Tasks

The TDO functional tasks were reviewed using information in Tables B-11 and B-12, filtering them through the user, organization, and situational context profiles described above. This process identified several key dimensions which defined task performance and error modes, including:

- task complexity and difficulty;
- task performance precision and accuracy requirements;
- input and feedback uncertainty; and
- task workload and potential stress dimensions.

It should be noted that probing task dimensions often triggers further refinement of the other profiles and all of this investigation involved repeated iteration in both top-down and bottom-up analyses. Figure 3.3 presents one of the conceptual maps created to describe the response requirements associated with real-time threats that "pop up" during ATO execution. The model, developed with Kaetron's TopDown® software, links the high-level problem aspects to additional models that address the details.

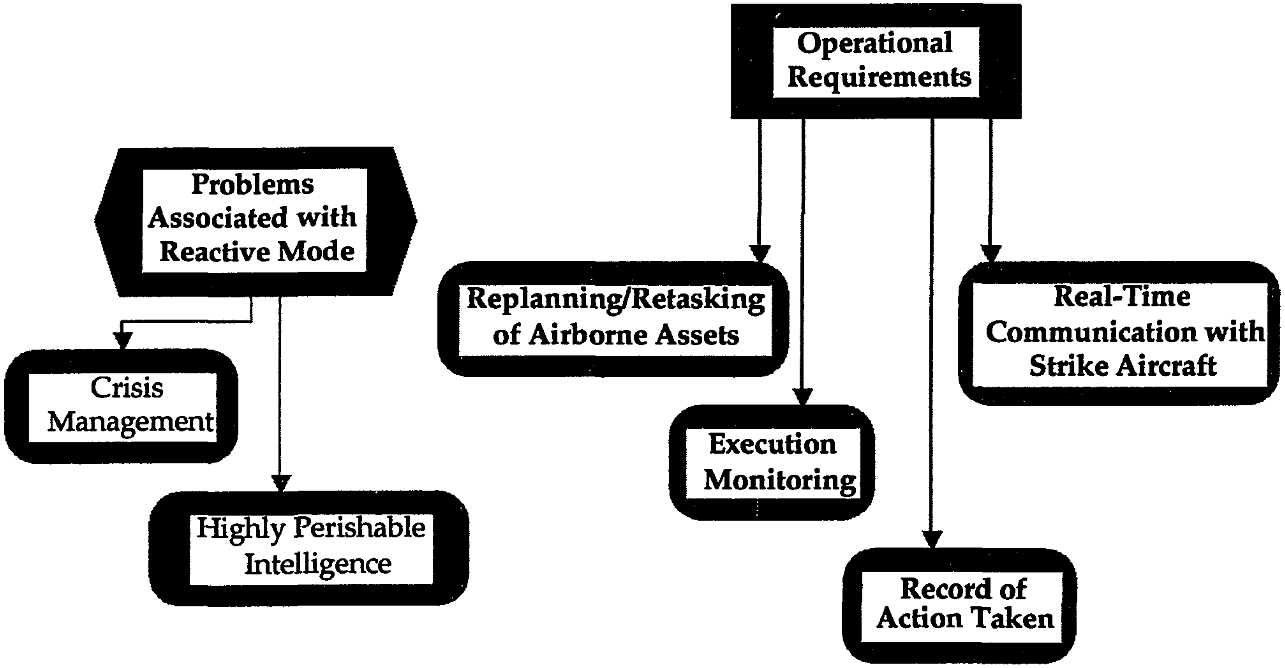


Figure 3.3: Response to Real-Time ("Pop-Up") Threats

Task Output

The TDOs' discrete output unit is the response to a task request for air refueling (AR) support. In a larger sense, the task output is also the overall status of the air refueling plan or the tanker operations system. The TDO is required to respond to a high volume of AR task requests as rapidly as possible; thus, they tend to be extremely intolerant of slow system response or highly complex routines for relatively simple tasks. Air refueling plans have multiple components and TDOs need system supports to prevent their losing track of all relevant plan components. For example, decision-makers need the ability to move through various levels of detail and system supports for structuring the various components to aid in analysis.

Task Response

The TDO's response goals are to meet the air refueling requirements of the ATO and maintain a viable air refueling plan for as long as possible. Both the short-term execution goals and overall mission completion goals are very difficult to attain. The system should be designed to offload the TDO of as much of the workload as possible (e.g., by allocation of table look-up and computational tasks to machine). Some of the subtasks (e.g., keeping track of taskable fuel) require high precision that is best allocated to the machine component. For example, the detailed data required for response precision can be maintained and manipulated by machine. In addition, automated updates relieve the TDO from being overwhelmed by the detail.

TDO response frequency during the execution of a major combat ATO is very high. As a result, AR tasks and changes to tanker operations pile up and must be prioritized to ensure the most important are handled as rapidly as possible. Delays in feedback (external or internal to COD) may impair the TDO's timely response.

Procedures and Subtasks

AR tasks arrive as discrete messages, but may have to be handled by considering the planning implications of several changes simultaneously. Handling a single AR task involves several steps, including the possibility of activating a ground alert tanker mission or creating a new tanker mission to resolve major changes to the AR plan. In addition, the TDO may have the current working task interrupted by a higher priority task. The requirement for the TDO to simultaneously handle the current AR tasks using FLEX while remaining a part of the off-line COD activity (e.g., incoming messages from other sources, conversations with other duty officers, etc.) also contributes to the time pressure experienced. The system must support the TDO's maintenance of situational awareness and task continuity, and complement the team activities of the COD.

AR subtasks are moderately dependent in terms of temporal order (either due to system or procedural constraints) and logical relationships; however, the subtasks are highly dependent with respect to the total AR plan. The overall dependency of AR plan is such that the complexity of relationships exceeds the TDO's ability to handle without support. The TDO needs a way to "step back" from the current situation to see the AR plan as a whole and understand the various direct and indirect dependencies. AR tasks' procedural complexity is moderately high to very high due to the number of subtasks potentially involved and the dependencies between them. Certain subtasks require strict adherence to set procedures; other subtasks may be handled in so many ways that a strict procedure is not prescribed. Where strict adherence to procedures is required, the system support must be designed to constrain TDO from ignoring critical procedures and make those constraints visible to the TDO. In contrast, where flexibility is allowed, the system should facilitate the TDO's ability to manipulate the options and make the affordances visible.

Task Input

Many of the input variables in the AR task are moderately predictable due the consistency of operational procedures, basic situational stability, etc. Some input values vary widely in predictability due to inaccuracy of supporting data or novelty of the situation. As a result, the TDO may need to be reminded of the less predictable aspects of the task to ensure that proper attention has been paid to the immediate contingencies ("what-ifs"). For example, variations which follow known patterns under certain conditions may be stored as templates to support faster recognition.

AR tasks are triggered in a very irregular fashion; the TDO generally cannot predict the flow of AR tasks with other than very gross metrics. The TDO cannot control the occurrence of the stimulus (AR task), but can control the order of response among tasks of the same priority. Although alarms may be shut off and incoming AR tasks acknowledged and set aside for later response, an AR task remains an open issue until changed by the TDO's response. Thus, the TDO may need to regularly review open requests and reorder priority under heavier workloads.

Task Feedback

More than 50% of the AR subtasks involve decisions based on feedback from previous responses. As suggested above, the TDO must respond to some high priority AR tasks immediately, while other tasks may be postponed temporarily. For this reason, the TDO needs to know when tasks will become critical to help in prioritizing numerous tasks with the same priority. Feedback to the TDO from other COD duty officers on actions taken is immediate; however, feedback from the tankers and other flying missions may be delayed by hours. As a result, feedback reference may be ambiguous as actions taken early in ATO day may be superseded by later events before feedback reaches the TDO.

As the ATO day progresses, TDO plan refinements may be entirely dependent upon the projected effects of plan changes for which there has been only partial feedback. The required reaction time for decisions is much less than the typical feedback lag and the TDO may have to make many dependent decisions long before feedback on one decision is received. This can result in over- or under-adjustments to the AR plan. To compensate, the TDO needs a means to model potential effects of actions against a likely model of the current situation. The secondary effects of feedback lag impact the effectiveness of the decision-maker's learning and experience. False assumptions due to feedback lag can generate inaccurate mental models regarding cause and effect relationships. For this reason, the TDO needs support for trying (and retracting) optional courses of action before committing to decisions.

3.2.5 Profiling the TDO's Decision-Making Tasks

The general characteristics of the FLEX functional tasks apply to all the duty officer positions. For this reason, most of the functional task identification described above was accomplished before the case study was narrowed to tanker re-planning operations. As the requirements identification shifted to the detailed profiles of the decision-making tasks, the focus narrowed to the Tanker Duty Officer (TDO) with particular emphasis on the decision-making activities involved in re-planning during ATO execution. Figure 3.4 presents one of the conceptual models developed to help identify the key activities and variables in tanker re-planning tasks.

This section presents the decision-making requirements identified and modeled using Tables B-12a-d in Appendix B. The TDO's cognitive task requirements are considered in terms of

- ***stimulus*** - situational input;
- ***hypothesis*** - situation interpretation;

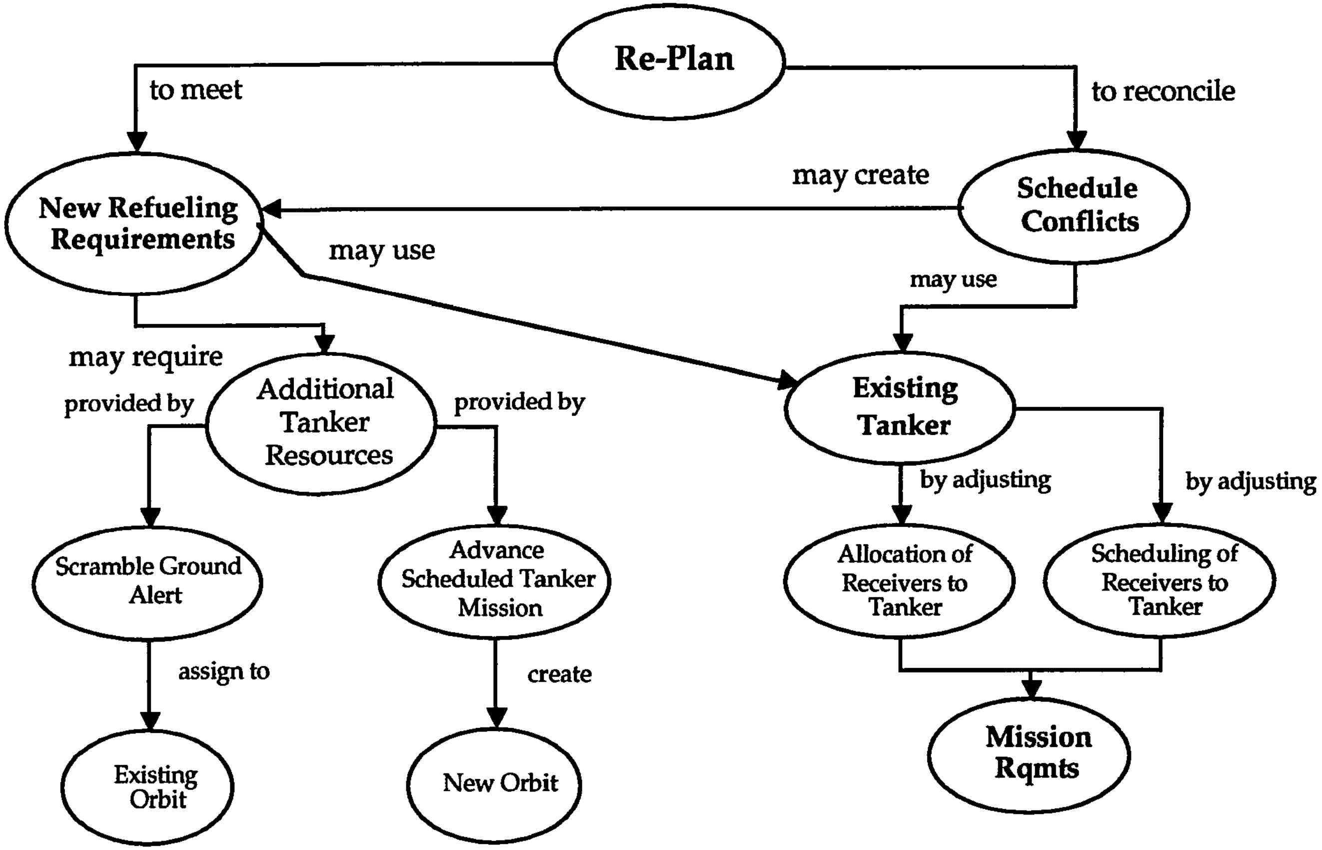


Figure 3.4: Re-Planning Tasks

- *option* - course of action review and selection, and;
- *response* - coordination and execution of chosen option.

Stimulus - Characteristics of the Situational Context and Data Inputs

Situation monitoring for the Tanker Duty Officer (TDO) in the Combat Operations Division (COD) is largely reactive. Unlike real-time tactical monitoring, the TDO is not directly manipulating the environment on a minute-to-minute basis. Instead, monitoring and decision-making are carried out in a time-constrained environment, primarily driven by incoming update alerts or task requests. Because important operations information may exist on multiple screens, the TDO needs to have changes brought to his attention. Pop-up display of new task requests makes detection of discrete air refueling (AR) requests automatic; however, the TDO may have considerable difficulty detecting underlying trends in tanker operations due to variations in the timeliness of updates to key variables.

Tanker operations information is primarily quantitative; qualitative information is inferred through maps and mission flows. The TDO's situational awareness requires supports for tailoring displays to filter, sort, and organize information. In combat situations, the volume of incoming updates to tanker operations data exceeds the human's ability to absorb or manipulate within the time requirements. The FLEX system automates the detailed updates and alerts the TDO to conflicts spawned by changes in resource availability.

FLEX information on tanker operations exists primarily as detailed data tables with summary information available in the Tanker Status Display Board.[5] The Map Graphic window charts information such as the locations of bases, tanker orbits and tracks, routes of planned missions, and defensive coverage.

---

[5] Windows from FLEX Prototype 3 are presented in Figures G-1 through G-7 in Appendix G.

FLEX users can filter the information presented to suit the requirements of their decision tasks. The Marquee is a graphic interface to much of the FLEX database. The Marquee's adaptable display presents some of the operational dependencies across the ATO timeline through a database feature that allows the user to sort and "bundle" dependent missions. However, the FLEX filtering does not adequately reduce workload due to complexity and information volume. Due to the screen layouts (particularly in the Tanker Worksheet), the TDO is still required to do some mental computation and make notes to keep track of certain variables. The TDO needs system support to reduce off-line mental computation and other memory requirements.

Tanker operations decision variables (e.g., fuel requirements, etc.) are generally understood and representative. When the required data are current, the variables are reliable for calculation and decision-making; however, this is not always possible due to communication failures or other feedback delays. Furthermore, the TDO may not fully assess the impacts of situation and options based on displayable information; there are potential "unknown unknowns" in combat operations which undermine the representativeness and reliability of standard decision variables. Mis-perception of the situation due to incomplete or ambiguous information can lead to any or all of the following:

- focus on irrelevant information;
- selection and/or fixation on an incorrect explanation or solution;
- incorrect interpretation of cues; or
- insensitivity to missing information.

Given these potential cognitive failures, the TDO may benefit from displays of system models or goal states to aid in

- identifying problems
- defining causal relationships;

- identifying missing information;
- interpreting ambiguous cues; and
- reducing over-confidence in decisions based on uncertain information.

The existing FLEX interface addresses some, but not all of these needs.

Hypothesis - Situation Assessment Task Characteristics

Several factors combine to make hypothesizing for situation assessment difficult. Although the TDO is familiar with all the activities of tanker operations, there is situational novelty inherent in the ways the variables may combine in combat. Joint service and multi-national (combined) operations add extra layers of complexity and novelty to tanker operations. Finally, the unpredictability of an intelligent adversary may result in an unfamiliar sequence of events. The combination of novelty with the crush of information flow may distract the TDO from seeing the underlying similarity to more familiar situations. To relieve the TDO, certain routine aspects of AR re-planning may be allocated to machine processes.

Situation assessment for air refueling operations is semi-bounded with a moderate number of hypothetical possibilities to explain current AR plan status; however, the number of hypotheses may seem greater under heavy workload situations. The TDO needs relief from complex detail through aggregated displays and interaction with models that help to identify the differences between the current and goal states. Goal-oriented displays of tanker operations also help to maintain focus on critical variables and serve as templates for analogies to familiar situations. Finally, to understand the potential direct and indirect effects of the current situation, the TDO needs a means of viewing the consequences of actions across the ATO day.

TDO performs situation assessment tasks in a time-critical, quasi-real time environment. This requires prioritizing backlogged tasks and often means trading off time to fully analyze situation in order to process more AR tasks in a shorter period of time. Comments for the FLEX Working Group (FWG) after all three prototype reviews indicated that the visual momentum involved in using FLEX was still relatively low due to the requirement to use operational data scattered across several windows to accomplish any task. To relieve the time pressure in situation assessment, the TDO needs "at-a-glance" displays that do not require hunting or elaborate manipulation of detail to get to the relevant information quickly. In addition, the TDO should not be burdened with off-line computation.

Most of the inferencing required for AR replanning is within set bounds, involving well-known parameters; however, the complexities of multiple receivers and their dependent missions creates a hidden network of inferences with varying degrees of certainty. This multi-dimensional network of inferences is very memory-intensive. To compensate, the TDO must use workload reducing heuristics that may introduce bias errors. The TDO needs displays which support inferencing based on accepted operational procedures. In addition, supports for option exploration should reduce the number of inferences and relieve the workload on TDO by portraying the current (and projected) state to compare with immediate and longer-term consequences across the network of tanker operation dependencies.

Option - Course of Action Decision Tasks

The number of possible options to a given air refueling (AR) situation are semi-bounded (as to the limits of available resources, etc.), but sufficient in number that the TDO faced with a large number of outstanding AR tasks is often overwhelmed by the resulting plan complexity. In addition, AR mission goals may shift several times in a relatively short period of time, requiring a re-evaluation of priorities, updates, and recalculation of projected changes in AR plans. Most of the conflicts and effects are predictable, but the number of conflicts spawned in interdependent missions by even a small plan change make manual manipulation intractable. Furthermore, the uncertainties and inherent complexity make outcome values for changing AR plans difficult to project despite the TDOs understanding of the fundamental variables.

The TDO needs facility to quickly package responses for less complex, more routine changes. The TDO needs some means of rapidly understanding the fundamental effects of an option under consideration. Ideally, the system display should support the decision-maker's rapid mental simulation to accept or reject the option as feasible. Although evaluating AR re-planning options is manually intractable under high workload situations, the problem is sufficiently bounded to allow for machine support in several areas, including:

- rapid recalculation of all dependent mission data to compare options;
- mapping of restructured dependencies; and
- highlighting any resulting conflicts.

To filter out the best option configurations, the TDO needs tools that allow rapid scoring of options against basic criteria with pre-determined or adjustable weighting. Where rankings are similar, the TDO needs displays that model or simulate the projected consequences for a given option to compare with other relatively equivalent options. Finally, the TDO needs to be able to step back from detail and view AR operations in terms of higher level goals. For example, predictable goal changes may be combined into contingency scenario templates and displayed to the TDO as advance notice or incorporated into a rule-based advisor.

Outcome uncertainty for most AR plan components is moderate, but predictable. Nevertheless, the broader the scope of the plan change, the less certain the outcome. TDO choices at time *t* may leave them more or less vulnerable at time *t* + 3. The potential vulnerability to later requirements changes (i.e., contingencies) is even more uncertain and difficult to factor into the decision. Combined levels of uncertainty add to the intractability of option evaluation. Moreover, feedback may not be timely, goals may change several times, and there is a very high penalty for making poor choices. The current FLEX system does not reflect the uncertainties aggregated into projected outcomes of AR plans. The system's ranking of options treats all quantitative data as being 100% certain. Thus, it is possible to have two equally ranked options, yet be unaware of their highly disparate levels of certainty. The TDO needs supports for understanding the degree of uncertainty inherent in a particular option.

Response - Planning, Coordination and Execution of Decisions

Air refueling plans are operational hypotheses involving multiple assumptions and inferences about the current situation and the causal relationships that predict outcomes. AR execution in high sortie ATOs can make use of pre-planned contingencies (e.g., by activating orbits and routes, launching ground alert tanker missions, selecting alternate recovery bases, etc.) to handle many of the plan changes. Extensive re-planning is required when major changes are made during execution (i.e., the addition of a large, high-priority mission; multiple failures; or resource losses). Re-planning decisions are further complicated by the difficulty of tracing all possible consequences of actions taken. The TDO needs support for decomposing new goals into AR subtasks and means-end restructuring of AR plans to meet new requirements.

Execution in tanker operations requires coordination with other DOs in the COD, with airborne forward control units, the affected strike wings and support operations. During joint and combined operations coordination also involves other services and national forces. AR coordination must take place within the decision horizon and is affected by the organizational shifts that occur in crisis conditions. Communication requirements for coordination (i.e., management of message traffic) impose processing loads on the system which constrain the design options. Reformatting to meet messaging standards qualitatively changes information passed and may affect its interpretation at the receiving end. Although coordination is handled through SODO and ATO distribution chain, the TDO needs support for understanding the potential coordination ramifications of options related to interdependencies and communication delays.

Execution of AR plan changes is a highly dependent, multi-phased control process. Multiple phases increase coordination requirements and can affect the feasibility of certain options due to the limits of the decision horizon. Delayed feedback may be incorrectly associated with the wrong phase and cause the TDO to over-correct. To track execution, the TDO might benefit from a display of goals and subgoals with current execution status.

3.2.6 The FLEX Cognitive Task Requirements (CTRs)

Appendix E presents a summary of the issues raised during the CTR identification phase for the FLEX Case Study. The goal of the requirements identification process was to re-examine the available requirements definition resources and enhance the existing FLEX requirements specification. Thus, many of the functional requirements identified are represented to some extent in the FLEX System/Segment Specification (SSS) and the FLEX prototypes. These high-level functional requirements for the HCI design group under three main support requirements: performance improvement, distributed decision-making, enhancing the decision-maker's knowledge base. Table 3.1 breaks these requirements down into their respective components.

Support for Improved Performance • Support rapidly adaptive response. • Provide DM most accurate, relevant information and technological means to combine and interpret information. • Offload DM of as much of the workload as possible. • Support pattern-matching, analogical reasoning, and other means for improving assessment in novel situations.
Support for Distributed Decision-Making • System must support the TDO's maintenance of situational awareness and task continuity, and complement the team activities of the COD. • Provide means to maintain overall control to meet mission objectives without direct review of every micro-decision by senior command • Optimize for fast communication to improve coordination and minimize authorization delays.
Support for Development of Decision-Making Knowledge • Make use of natural or domain knowledge in the interaction symbology to allow the user to interact with the task in the most familiar terms. • Display structural information (i.e., functional cause and effect relationships) to aid development of mental models and support wider knowledge of response options. • Provide doctrinal/procedural overview displays to support interpretation of and effective response to novel or rare events. • Provide varying levels of explanation to support the construction of more robust mental models.

Table 3.1: High-Level Functional Requirements for the HCI Design

3.2.7 Specific Cognitive Task Requirements

Appendix E presents a complete list of the cognitive task requirements and related issues raised during the requirements identification phase. It was necessary to narrow the scope of the Tanker Re-Planning Case Study to three key CTRs, unrepresented in the FLEX SSS and unmet in the FLEX Prototype 3. These included requirements to

1.) adjust the problem viewpoint (level of detail),
2.) focus attention on the key decision variables, and
3.) compare response options in terms of potential consequences.

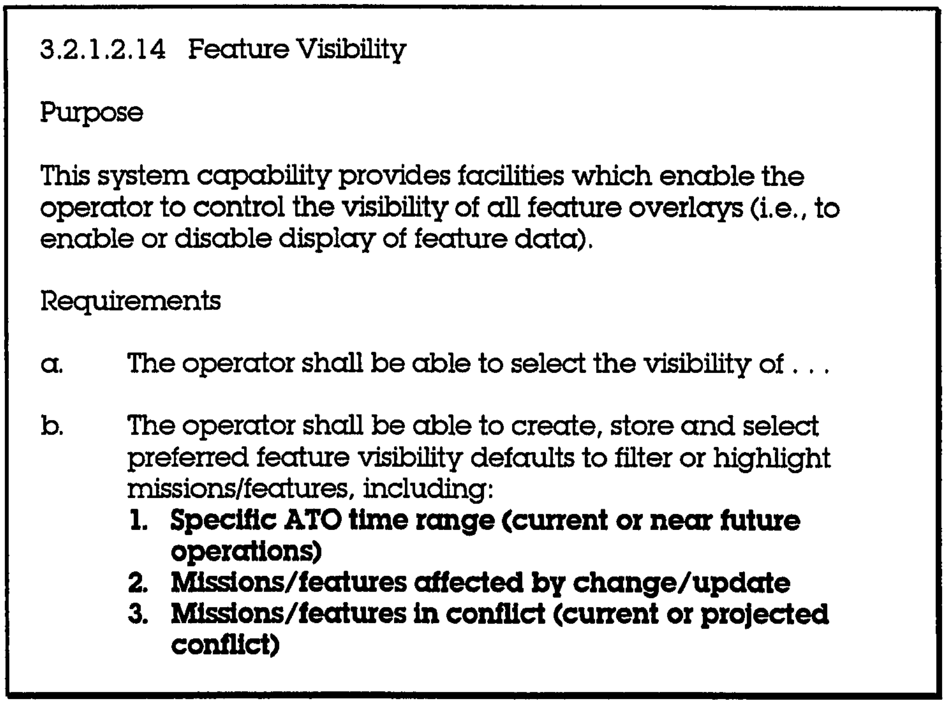
First, the TDO needed a way to "step back" from the detailed data with an overview of tanker operations. This was, in part, a response to the time horizon of the TDO's decisions and the varying degrees of timeliness and precision connected with the updates to the database. Small changes to the published ATO which must occur rapidly (e.g., last-minute re-routing of a mission to another tanker for refueling) are handled in the air by forward controllers. The TDO makes decisions involving a somewhat longer decision horizon and needs to work with an aggregated display of the entire ATO day. Second, the TDO needed a display simultaneously presenting all the critical decision factors. The working group participants complained that key information was distributed across several displays, requiring the user to jump around and make notes off-line. Finally, the TDO needed a support for mentally simulating the chain of consequences (e.g., changes in critical values) associated with feasible options. Answering these requirements without sacrificing access to detail became the central goal of the CSE interface re-design.

The complete list of cognitive task requirements presented in Appendix E was integrated into the FLEX System/Segment Specification (SSS) as described in

Section 3.3. It was also used to distill the design goals for the CSE-based HCI prototype (Section 3.4).

3.3 Integrating Cognitive Task Requirements into the System Requirements Document

The Department of Defense development standard for software systems, DOD-2167A, specifies the format and content of system-level requirements documented in a system/segment specification (SSS) document. Although the FLEX case study focused on the decision activities of the Tanker Duty Officer, the CTRs had to be identified and represented in the higher level format of the FLEX SSS. This integration involved distilling the findings from the requirements review presented in Appendix E and matching them to the relevant system specifications in the existing FLEX SSS. In many cases, the FLEX SSS already contained statements which incorporated the content of the CTR. Occasionally, the statements were modified to improve their precision. In addition, items were appended to stated requirements to detail functionality specified by identified CTRs. Figure 3.5 provides an small example from the amended FLEX SSS. Appendix F presents extended examples from the integrated System/Segment Specification (SSS) for the FLEX Case Study.



3.2.1.2.14 Feature Visibility

Purpose

This system capability provides facilities which enable the operator to control the visibility of all feature overlays (i.e., to enable or disable display of feature data).

Requirements

a. The operator shall be able to select the visibility of . . .

b. The operator shall be able to create, store and select preferred feature visibility defaults to filter or highlight missions/features, including:
   1. **Specific ATO time range (current or near future operations)**
   2. **Missions/features affected by change/update**
   3. **Missions/features in conflict (current or projected conflict)**

Figure 3.5: Example of a CTR Integrated in the FLEX System/Segment Specification Document (added CTR appears in bold face)

3.4 Translating Requirements to an HCI Design Concept

Design Goals

The cognitive task analysis presented in Section 3.2.5 repeatedly raised certain cognitive aiding issues. These cognitive aiding requirements aggregate into categories of design goals representing situational awareness and understanding, attentional focus, reduction of mental workload, problem perspectives, option evaluation, decision control and guidance, interface operation and error control (Appendix E, Section 6.0). The last two goals involved requirements that were adequately addressed in the existing FLEX prototype and lay outside the specific interests of this research. The remaining six belong to the general category of improving decision-making performance represented in the three FLEX CTRs listed in Section 3.2.5. These requirements were addressed to some degree in the FLEX SSS and the FLEX prototype designs. Each is re-capped briefly below.

Goal 1: Support for Situational Awareness and Understanding

- Provide display features (e.g., overview screens) to help the user develop mental models of the operational environment.
- Make the sources and extent of uncertainty explicit.
- Provide templates of various known patterns and causal conditions to support faster recognition.

Goal 2: Support for Focus on Goal/Decision-Relevant Information

- Provide goal- or decision-oriented displays to focus attention on relevant information and support
  - » identifying the situation and/or problem;
  - » defining causal relationships;
  - » identifying missing information;
  - » interpreting ambiguous cues; and
  - » reducing over-confidence in decisions based on uncertain information.
- Provide predictable goal changes in contingency scenario template displays.

Goal 3: Support for Understanding of Operational and Domain Dependencies

- Provide system-level (i.e., tanker operations) displays to convey interdependencies and situational overviews.
- Example: the TDO needs ability to display integrated tanker-receiver dependencies, mission flows on all active tanker orbits and fuel available.

Goal 4: Support for Reducing Mental Workload

- Provide system support to reduce off-line mental computation and other memory requirements.
- Provide an option to use supports (e.g., table look-up tasks) and reminders.
- Provide and propagate automated updates to relieve the TDO of the overwhelming task of maintaining detail.

Goal 5: Support for Viewpoint Adjustment

- Provide the TDO the ability to adaptively filter information to permit the required abstraction level, while retaining rapid access to detailed information.
- Provide the ability to "step back" from detail and view AR operations in terms of higher level goals and the various direct and indirect dependencies.
- Provide "at-a-glance" displays that do not require hunting or elaborate manipulation of detail to get to the relevant information quickly.

Goal 6: Support for Option Comparisons

- Provide a means of viewing the consequences of actions (including the indirect effects) across the ATO day.
- Provide support for trying (and retracting) solutions before committing to decisions.
- Provide a means for a rapid mental simulation to accept or reject the option as feasible.
- Provide displays which support inferencing based on accepted operational procedures.

- Provide support for rapid scoring of options against basic criteria with pre-determined or adjustable weighting.
- Provide displays that model or simulate the projected consequences for a given option to compare with other relatively equivalent options.
- Provide support for understanding the degree of uncertainty inherent in a particular option.

In addition to the immediate benefit of improving performance, Goals 1 - 3 have the potential to enhance long-term performance by developing and reinforcing the mental models that produce a more robust decision-maker knowledge base.

The FLEX Tanker Case Study focused on the immediate benefits of performance improvement derived from the six design goals. Figure 3.6 maps the interdependencies associated with the individual goals. Research indicates that the quality of situation assessment and ability to preview the effects of decisions improves decision performance (Klein *et al*, 1992; Klinger *et al*, 1993; Raphael, 1991). In particular, improving the DM's understanding of the causal dependencies that underlie a situation and the consequences of a given course of action can help to reduce decision error often associated with complex decisions (Cohen *et al*, 1985; Reason, 1990; Senders and Moray, 1991). The keys to situational awareness and understanding lie in the DM's ability to

1) filter the relevant situational cues from the complex barrage of data, and
2) combine the cues to make inferences about the situation (Andriole and Adelman, 1989).

Selecting the appropriate level of detail and focusing on decision-relevant information assists the filtering process; while an understanding of the operational and domain dependencies -- the causal networks -- provides a framework for combining information to make inferences. Relieving the DM of certain detailed

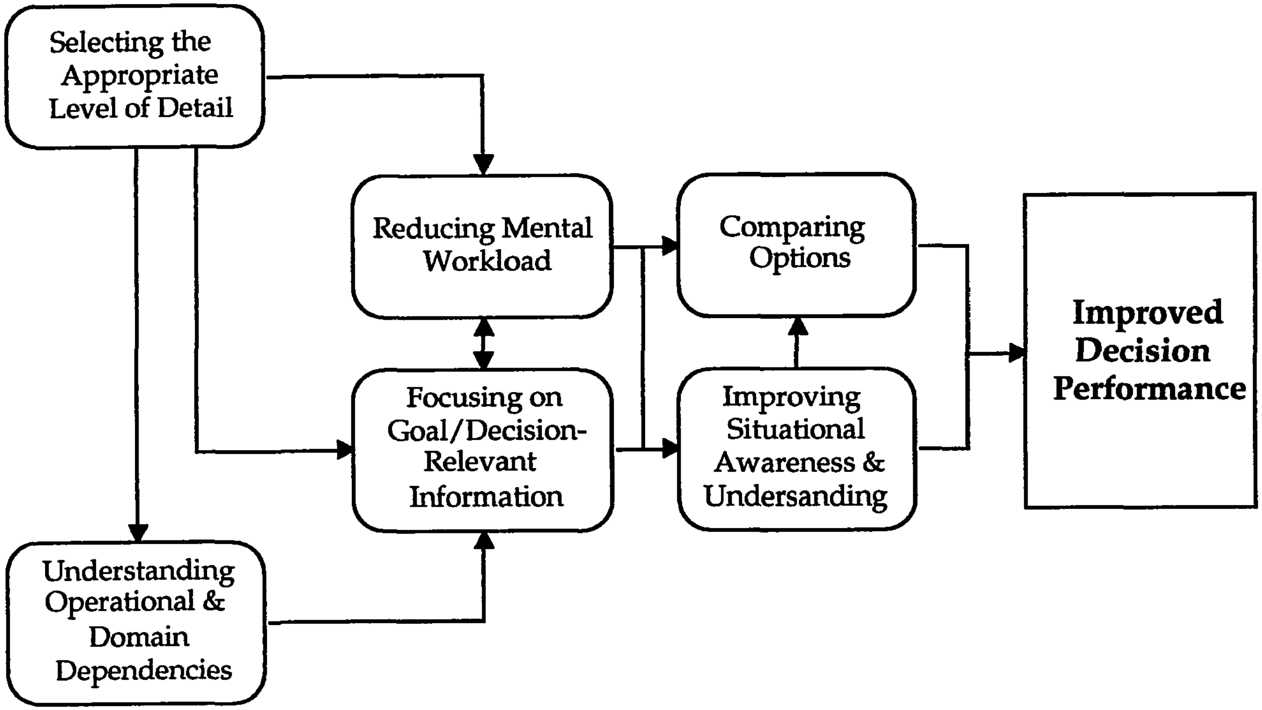


Figure 3.6: Relationship of FLEX Design Goals to Overall Goal of Improved Decision Performance

mental operations (e.g., calculations, table look-up operations, and various memory tasks) and providing mental organizers (e.g., decision-structured displays) permits the focus of mental resources on the critical decision tasks. Finally, the ability to compare options in terms of potential consequences of actions taken is enhanced by the DM's focus and understanding.

The CTRs identified for the FLEX Tanker module during the requirements identification phase and incorporated into the six design goals above map to four CSE design principles. These principles, with the associated design goals in parenthesis, include:

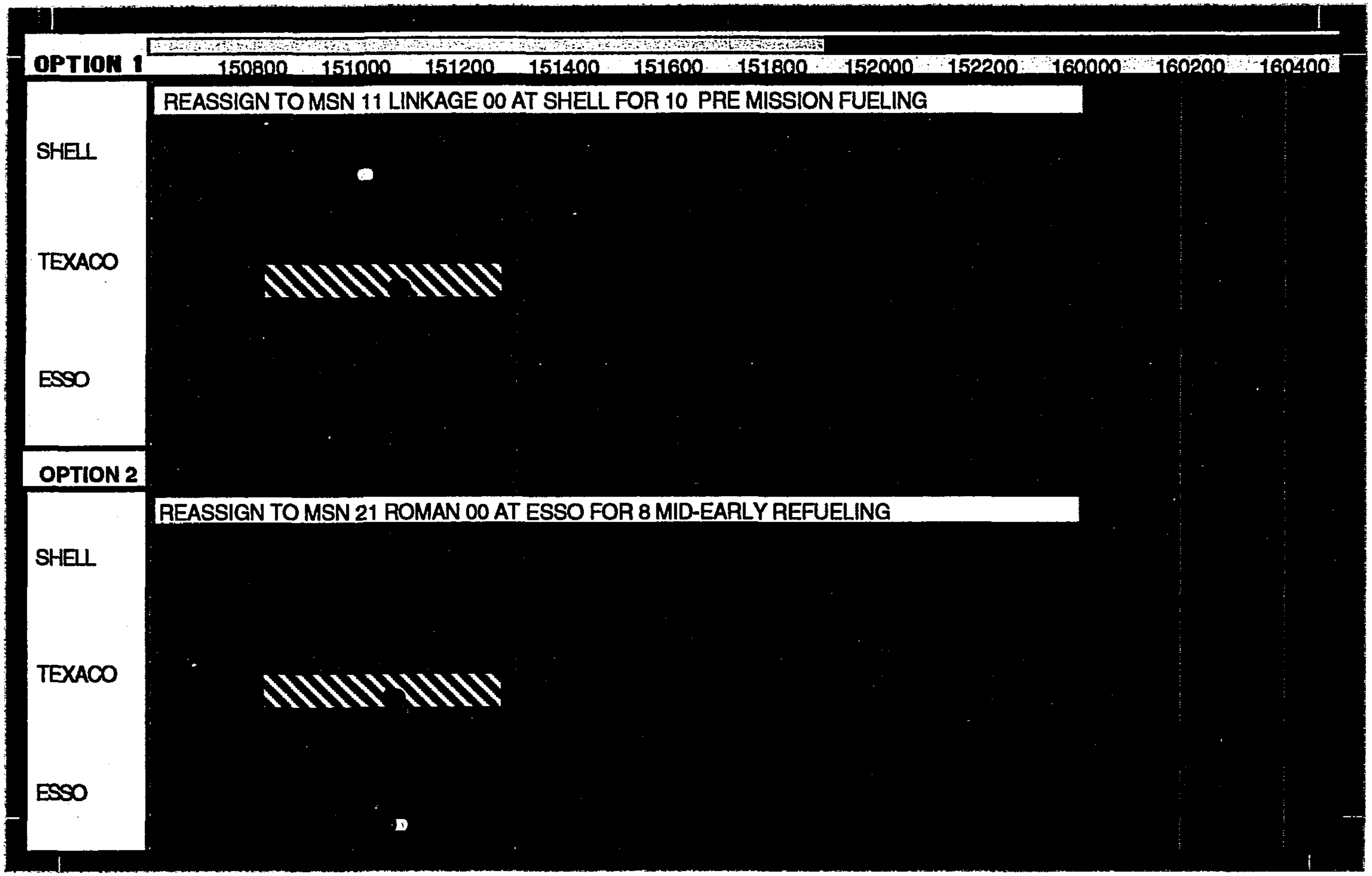
- Presenting a system-level model relating the relevant decision variables to focus the decision-maker's attention and guide the selection of appropriate detail to further inform the decision process (Goals 1 - 6);
- Integrating all the key decision factors in one display to eliminate unnecessary jumping from screen to screen (Goals 2 - 5);
- Making the current system (i.e., tanker operations) state visible to highlight the areas requiring correction (Goals 2 - 5);
- Relieving the DM of calculation and memory tasks (Goals 2, 4 and 6);
- Making the consequences of options visible for comparison and evaluation (Goals 1, 2 and 6).

The first two principles were drawn primarily from the ecological interface design research by Jens Rasmussen and his colleagues (Rasmussen and Vicente, 1989; Vicente and Rasmussen, 1992) and represented in guideline form in Rasmussen and Pejtersen (1993) and Rasmussen *et al*, (in press). In addition, research on the design of integrative displays (Bennett *et al*, 1993) provided further insight into the ways decision cues can be combined in symbolic displays whose decision-aiding "emergent" features are only apparent in that combined form. Finally, the tactical decision-making research by MacMillan and Entin (1991) illustrated the decision performance value of unifying the key decision factors in a single window. The three remaining principles reflect guidance that may be found in all standard guideline sources.

The guidance from these principles drove the design of an additional window for the FLEX Tanker DO called ***Option View***. (Figure 3.7). The ***Option View*** window incorporates a number of HCI responses to the CSE principles identified. First, the window presents a high-level system model of current tanker operations displaying the active tanker missions at their orbit locations across the 24 hours of the ATO. The receiver contacts are mapped across time against the assigned tanker mission to highlight their flow in terms of density and timing. Conflicts are highlighted in red to draw attention; changes in the tanker or receiver missions are highlighted in yellow. The taskable fuel remaining is displayed above each tanker mission and relieves the DM from having to make the calculation. Second, to facilitate comparison, two options may be compared simultaneously against the planned ATO. (The actual large-screen monitor used for the Air Force FLEX prototype would support comparison of more than two options.) The comparisons present the effects of allocations in terms of changes to the taskable fuel remaining, timing of receiver contacts, and density of assigned receivers against the tanker.

3.5 Developing an Interactive Prototype of the HCI Design Concept

The FLEX ATTD is a technology demonstration program that is intended to evolve into a fielded system. Given the author's external role in the FLEX ATTD, the FLEX Tanker Case Study made use of a throwaway prototype to evaluate the HCI design impacts on decision performance. For evaluation and comparison, both the FLEX tanker module displays and the revised HCI design were imple-



OPTION 1

REASSIGN TO MSN 11 LINKAGE 00 AT SHELL FOR 10 PRE MISSION FUELING

OPTION 2

REASSIGN TO MSN 21 ROMAN 00 AT ESSO FOR 8 MID-EARLY REFUELING

Figure 3.7: Option View

mented in an interactive prototype. The essential features of the existing FLEX windows were mocked-up to allow for rapid prototyping of the key decision factors presented in each window (Appendix G). The extensive searching, sorting and tailoring capabilities of these displays were not represented in order to focus the evaluation on the decision-making tasks rather than the interface manipulation tasks. The evaluation prototype was developed in SuperCard® on an Apple Macintosh IIci® with a high-resolution RGB color monitor. To facilitate non-intrusive, automated data collection, the software program includes routines to record time-stamped information about the user's interaction with the interface.

3.6 Evaluating the HCI Design Concept

In rapid prototyping development efforts, software evaluation goes on continuously as functional modules are developed and integrated. In similar fashion, HCI concepts and features may be evaluated early in development as design hypotheses. Such early evaluation is particularly important when the system contemplated will comprise a major change to the decision-making organization. Early concept evaluations are also useful for evaluating the value-added by incorporating advanced HCI technologies.

In addition, to the narrowly focused evaluations conducted throughout the life-cycle, the overall HCI design must be evaluated as part of a total prototype evaluation. This allows the designers to examine the flow of interaction between the user and the computer and explore interface problems that may not surface in limited studies. Overall evaluation is best conducted using subjects that represent a cross-section of the target end-user population. Although the FLEX Case Study only focused on a small subset of the larger FLEX system, the case study evaluation was conceived in terms of a complete review of the HCI concept in the prototype. The remainder of this section briefly describes the evaluation goals and design. The evaluation is presented in detail in Chapter 4.

3.6.1 Developing Evaluation Goals

The fundamental hypothesis of the cognitive systems engineering framework is that using the approach should highlight the critical cognitive task requirements and, by guiding the translation of these requirements into design concepts, result in changes in the system which, in turn, result in changes in task performance. The evaluation of the FLEX Tanker Module Prototype sought to validate the CSE approach by demonstrating an improvement in decision performance along three dimensions: situational awareness and understanding, option evaluation, and cognitive workload. These dimensions incorporated the six design goals identified in Section 3.4.

3.6.2 Selecting Evaluation Methods

The evaluation goals identified were very specific to the cognitive task requirements and unique features of the tanker operations domain. For this reason, it was critical to evaluate the task interaction concepts as well as the information presentation aspects of the HCI design. The RL version of the FLEX prototype did not have facilities for setting up multiple small trials. More importantly, the interface was both "fragile" (i.e., prone to frequent crashes) and very difficult to learn. The prototype developed for evaluation focused on the decision tasks and minimized system operation tasks by pre-formatting the highly customizable FLEX windows so that any window called by the user would display its information to best advantage. This was done to eliminate performance variation due to differences in system operation skills. The high level of domain and task knowledge that characterized the target users suggested that subjects for the interaction should be drawn from a Air Force officers with a common level of knowledge and experience in tanker operations.

As indicated previously, the framework for the evaluation of the FLEX HCI design was built upon a multi-dimensional view of the factors contributing to effective decision-making performance. The fundamental hypothesis for evaluation may be stated as follows:

> HCI designs based upon the CSE approach to identification and specification of cognitive task requirements will result in improved decision-making performance.

This high-level hypothesis was broken down into measurable factors with respect to three dimensions: situational awareness and understanding, option evaluation and selection, and cognitive workload. Each dimension was represented by one or more design goals that, in turn, were the subject of one or more sub-hypotheses and measures. Figure 3.8 maps the six evaluation hypotheses and related measures to these three dimensions. Each dimension is discussed in turn below.

Dimension 1: *Situational Awareness and Understanding*

- Design Goal: The presentation of information was designed to highlight and relate key decision factors at the appropriate level of abstraction to relieve DMs from the requirement to accomplish this integration in their heads.

  Hypothesis 1.1a: Decision-makers presented an integrated model of the "system" and critical decision variables will more accurately focus their information search than those not supplied with the integrated model display.

  Hypothesis 1.1b: In the absence of a fully integrated model display, decision-makers will compensate by selecting the displays which partially integrate key variables.

  Measures:

  1. Time-stamped Process Trace of Information Views Used (comparison with decision model of where critical decision information is located)

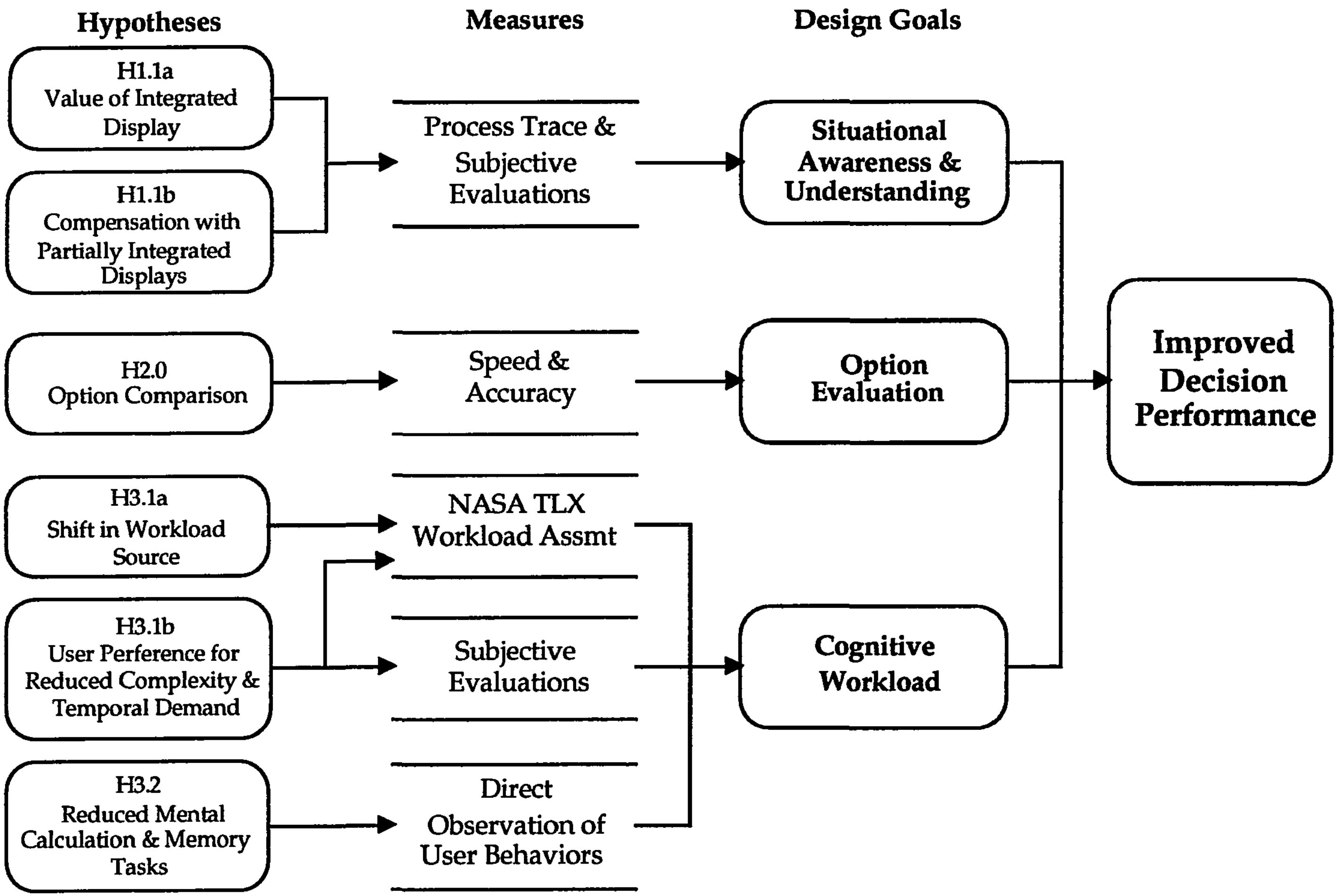


Figure 3.8: Relationship of FLEX Evaluation Hypotheses & Measures to the HCI Design Goals

- comparison of mean frequency of window selection
- process trace (precisely where user went when)
- comparison of mean duration (seconds) spent viewing each window

2. Subjective Interface Evaluations

   (comparison of interface/task means based upon users rating on discrete scale of specific window's usefulness in four decision tasks)

   - Problem Identification
   - Situation Assessment
   - Option Evaluation
   - Option Selection

Dimension 2: Option Evaluation

- Design Goal: The information presentation and interaction was designed to allow exploration and comparison of two or more options in terms of their consequences across time.

  Hypothesis 2.0: Displaying the changes in the critical variables to allow simultaneous exploration of two or more options will improve option evaluation and selection performance.

  Measures:

  1. Speed (comparison of mean times to make individual decision - trial and sum - by interface)
  2. Accuracy
     - comparison of mean score on selection of "better" option across trials, users, and interfaces
     - comparison of ANOVA on scores across trials, users, and interfaces ("better" option determined by previously established experts' model rating options based on taskable fuel remaining and receiver "density" function)

  NOTE: Interface exposure order effects were compared to evaluate the potential task and interface learning interaction across sessions.

Dimension 3: *Cognitive Workload*

- Design Goal: Reduce the users' experience of cognitive workload due to mental demand and time-pressure by designing the information presentation as a "system model" representing and relating critical decision variables.

  Hypothesis 3.1a: When other task factors are held constant, the perceived workload associated with time-pressure and problem complexity will be greater for decision-makers working without integrated displays.

  Measure: NASA-TLX workload assessment.[6]

  - comparison of the percentage of total workload attributed to temporal and mental demand depending upon interface used

  Hypothesis 3.1b: The subjective evaluation of interfaces will favor those interfaces associated with lower cognitive workload ratings (i.e., those that reduce task complexity in terms of mental and temporal demand).

  Measures:

  1. NASA-TLX workload assessment
     - mean percentages by interface
     - mean total workload by interface
  2. Subjective Interface Evaluations

  - comparison of mean subjective evaluations interface effectiveness across decision tasks (problem identification, situation assessment, option evaluation, option selection)
  - review of open-ended written and verbal impressions of interfaces (audio recording of discussion after final session) vis-à-vis task requirements

---

[6] NOTE: NASA TLX is a subjective rating of the user's perception of the source of task workload across multiple dimensions (e.g., mental demand, temporal demand, own performance, frustration, effort, etc.) Further details are presented in Chapter 4.

- Design Goal: Display the changes in the critical variables to relieve the decision-maker of the extra cognitive workload involved in mentally simulating the comparative effects of the options. Allocate tasks, such as calculation of numerical values (e.g., fuel remaining), to the computer to relieve users of mental calculation.

  Hypothesis 3.2: Decision-makers provided integrated displays (i.e., those presenting calculations of all key variables) for comparing the options will not make off-line notes to support their mental simulations.

  Measure: Direct observation - collection of session materials for review (i.e., did the users make notes and calculate values while using the interface)

The prototype evaluation was specifically designed to explore the constructs behind the cognitive task requirements and demonstrate the range of information that could be gathered and analyzed quickly. The various measures selected were chosen for their presumed validity as measures of the criteria of interest, but preference was given to methods that were either very quick to analyze or could be automated in the software of the interface prototype. For example, process measures were chosen which could be captured and compiled automatically rather than employing a team of observers, transcribers, and coders to collect and format verbal protocols. The subjects were provided several opportunities to comment on the nature of the interaction and the information presentation. As much as possible, these subjective data were collected in structured formats that facilitated rapid coding and analysis.

Since a sufficiently large group of representative users is difficult to obtain for long periods of time, the evaluation sessions were designed to require each participant to commit to only two half-day interaction sessions. Counter-balanced exposure and a repeated measures design provided sufficient power to achieve significant results with a total of twelve subjects.

The results of the evaluation generally supported all hypotheses. Comments from the subjects after exposure to both interface designs strongly favored the CSE-based addition of the ***Option View*** window. Moreover, the subjects' difficulties with the tasks when using the original FLEX interface conformed to the errors predicted during the requirements identification. Full details of the experimental design, subjects, procedure and results are presented in the next chapter.

4. Evaluating the Cognitive Systems Engineering Framework

Evaluating the CSE framework requires examining the *benefits* of the approach in terms of the output, or products, of the system development process versus the *costs* in development time and resources. Modeling and incorporating the human decision-maker's requirements in a human-computer cooperative decision-making system should improve the quality of the overall system requirements identification which, in turn, should support system designs that improve decision-making performance. Figure 4.1 presents a breakdown of the evaluation of CSE design processes and products involved in the FLEX Tanker Operations Case Study. The first section in this chapter presents the experimental study performed to evaluate the benefits of the design changes with respect to decision-making performance and processes. The second section follows with the evaluation of the CSE framework in terms of the potential costs of incorporating the method into the development process and the benefits in reducing design rework. The chapter ends with a summary of the evaluation findings and areas for further research.

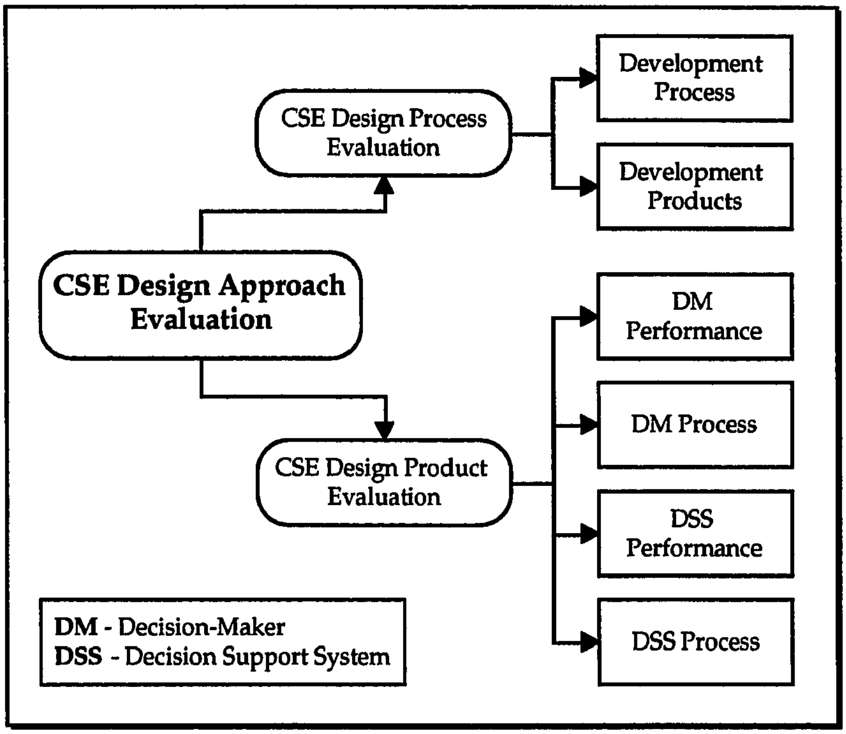


Figure 4.1: Organization of the CSE Framework Evaluation

4.1. Evaluating the CSE Design Product

This section presents the experiment conducted as part of an evaluation effort designed to assess the potential benefits of the CSE framework for developing systems that improve decision-making performance. Figure 4.2 breaks the product evaluation into the objective and subjective measures of effectiveness for the performance of the decision-makers and the decision aiding system. The subsections present the experiment as follows: 1) the experimental design and hypotheses, 2) the experimental testbed, 3) detailed descriptions of the two system interfaces, 4) the experimental procedures, 5) the dependent measures, and 6) the results of the experiment.

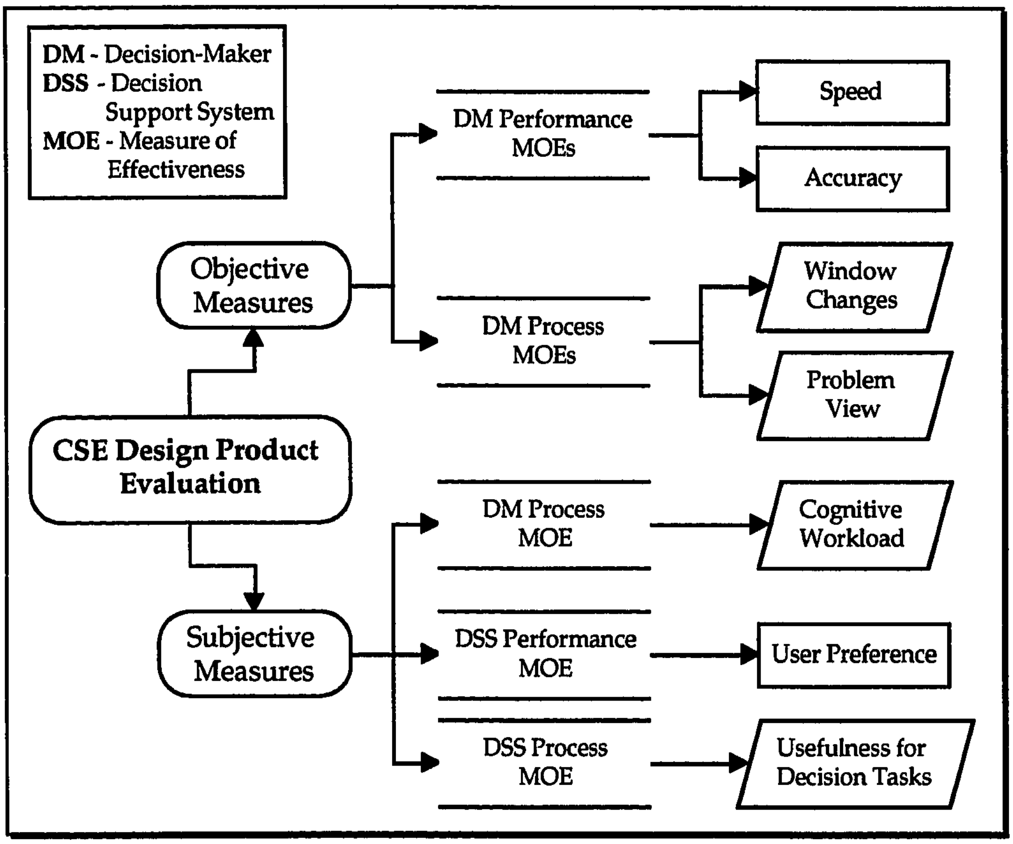


Figure 4.2: Organization of the CSE Design Product Evaluation

4.1.1. Design and Hypotheses

The evaluation design employed a 2 (interface) by 2 (exposure order) factorial design. The system interface was a within-subject variable, with all subjects interacting with both interfaces. Exposure order was a between-subject variable, counter-balanced to allow for the potential learning effects associated with interface and task exposure. The two system interfaces employed were:

a) the original Force-Level Execution (FLEX) system interface, as developed in Prototype 3 by the development team at the Advanced Concepts Branch, Rome Laboratory, Griffiss AFB, NY; and

b) the CSE system interface, adding the ***Option View*** window to the FLEX system interface described above.

The CSE interface prototypes one possible improvement to the original interface based upon the application of CSE framework for requirements identification and design.

The FLEX system interface was designed to support Duty Officers (DOs) in the performance of a variety of missions and tasks in the Combat Operations Division (COD) during the execution of an Air Tasking Order (ATO). The experiment and the CSE interface design, focused on the Tanker Duty Officer's (TDO's) replanning tasks, particularly the adjustment of tanker-receiver rendezvous assignments due to changes in the ATO or combat situation. The experimental testbed and the two system interfaces are described in the next subsection.

At a conceptual level, the CSE system interface differed from the original system interface in its focus on decision-oriented versus data-oriented information display and interaction. This difference was hypothesized to improve the cognitive processes involved in decision-making tasks and, therefore, improve overall performance on outcome measures.

The principal interface feature changes provided by the addition of the ***Option View*** window and predicted effects include the following:

1. ***Option View*** presents tanker operations (all tanker missions and associated orbits/tracks) across the entire ATO "day" as a system of tanker resources and dependent receivers. This system-level overview was predicted to augment two cognitive processes:
    a) the TDO's understanding of the current situation and trends in the evolving situation; and

   b) the TDO's evaluation of possible courses of action (COAs) and their potential effects on tanker operations and the accomplishment of the ATO missions.

2. *Option View* provides graphic and alpha-numeric representation of the key decision variables for replanning tanker-receiver assignments (i.e., taskable fuel remaining, timing and density of receivers assigned to tankers). This feature was predicted to improve the decision-maker's cognitive processes in the following ways:
   a) enhancing the TDO's decision performance by vividly presenting the key decision variables in a single location;
   b) reducing the TDO's cognitive workload by eliminating manual calculation tasks and reducing the workload associated with task complexity and time stress;
   c) focusing the TDO's information search by allowing the decision-maker to determine the appropriate format for information display.
3. *Option View* allows the decision-maker to simultaneously compare two options and preview their subsequent effects on the tanker operations system. This feature was hypothesized to improve the TDO's option evaluation and selection processes by permitting the decision-maker to compare the short- and long-term effects of an option in terms of the key variables and the operational interdependencies.

Twelve subjects were drawn from tanker squadron officers in the 509th Aerial Refueling Squadron at Griffiss AFB, NY to serve as TDOs for the experiment. All subjects (junior aircraft commanders and senior co-pilots) had an equivalent level of experience and training in tanker operations. Although all officers had flown missions in the Mediterranean and the Persian Gulf, none of the participants had experience in force-level planning. The volunteer subjects were randomly assigned to one of four groups by the squadron operations center based upon scheduled availability.[1]

Subjects in the four groups participated as individuals in one interactive session with each interface. To control for interface learning effects, the order of interface exposure was varied by assigning two groups to each experimental block (Table 4.1). Each interface session consisted of 12 successive decision trials to permit a repeated measures design. All subjects completed both sessions as scheduled providing a complete data set for the experimental analysis.

INTERFACE	INTERFACE EXPOSURE ORDER	
	Original-1st	**CSE-1st**
Original	Session 1	Session 2
CSE	Session 2	Session 1

Table 4.1: FLEX Case Study Data Collection Design

4.1.2. Experimental Testbed

The experiment was conducted during September and October 1993 at Rome Laboratory, Griffiss AFB, NY. For evaluation and comparison, the tanker module of the original FLEX interface and the CSE-based interface were implemented in interactive storyboard prototypes. These evaluation prototypes ran on Apple Macintosh® computers with high-resolution RGB color monitors. To facilitate non-intrusive, automated data collection, the software program included routines to record time-stamped information about the user's interaction with the

---

[1] Four groups were required due to the limits of the testing facility which could handle only three subjects per session.

interface. Figures 9-11 in Appendix G present the screens associated with this aspect of the data collection. These windows include:

- ***Subject Card*** - records subject data, trial times and option chosen;
- ***Subject Tracker*** - records subject interaction for each trial; and
- ***Option Select*** - allows user to input their option choice at the end of each trial.

Twelve experimental trials were developed with scenario input from several Air Force officers with recent experience in tanker operations. Specific data details for the trials (e.g., fuel requirements based on aircraft type, mission and route) were generated by the Automated Planning System (APS), a stand-alone operational prototype developed to support the Combat Plans Division of the Air Operations Center (AOC) in mission planning and ATO generation. When operationally deployed, the FLEX system will receive planned ATOs for monitoring and execution from APS. The "best" option for each trial was identified in coordination with this team of officers as an expert judgment.

4.1.3. The Two System Interfaces

The essential features of the existing FLEX windows were "mocked-up" to allow for rapid prototyping of the key decision factors presented in each window. The extensive searching, sorting and tailoring capabilities of these displays were not represented in order to focus the evaluation on the decision-making tasks rather than the interface manipulation tasks. The original and CSE interface windows, presented in detail in the previous chapter and reproduced in Appendix G, may be summarized as follows:

- <u>Original FLEX Tanker Module Windows</u>
  - » ***Task Notify*** - alerts DO to new task assignment;
  - » ***Task Inspector*** - provides summary data about pending task;

  - » ***Tanker Worksheet*** - presents details on missions, fueling requirements, tanker availability by base, etc.;
  - » ***Tanker Status Display Board (SDB)*** - presents data on the status of planned tanker missions;
  - » ***Map Graphic*** - displays a map of the battle area indicating tanker orbits, mission routes, and associated information;
  - » ***Marquee*** - displays mission flows across time based on user queries; and
  - » ***Replanning Options*** - presents a ranked set of replanning options.
- CSE Tanker Module Windows
  - » All of the windows in the Original interface plus
  - » ***Option View*** - graphic view of tanker operations showing active and ground alert tanker missions with their associated refueling receivers across the ATO day.

Both interfaces were operated in identical fashion with a mouse-based graphic user interface. The interface featured pull-down menus and "clickable" buttons. Windows were selected using either the menu bar menu of Tanker Windows or clicking on the desired window with the mouse. All windows could be re-sized (by cropping) and moved to any location on the screen. Windows could also be pushed to virtual screen locations off the edges of the visible screen and returned by dragging or selecting them from the menu bar menu.

4.1.4. Experimental Procedures

Each subject participated in two 3-1/2 hour sessions at the testing facility set up at Rome Laboratory. In the first session, the participants completed background information forms and signed the informed consent forms.[2] Following this, the subjects received instruction on the task domain (force-level re-planning

---

[2] Copies of all data collection forms are contained in Appendix I.

decisions), fundamental processes of decision-making, the NASA TLX workload assessment forms, and the interface for that session. Before the experimental trials began, subjects practiced for 20 minutes on an example trial using the interface.

Task domain instruction described the activities of the COD and the duties of the TDO as part of the COD team. The presentation then focused on the external and internal factors that trigger breakdowns in the planned execution of tanker missions. Discussions emphasized the relationship of these factors to such variables as taskable fuel available, refueling equipment (e.g., boom or drogue), speed and altitude, tanker location with respect to receiver mission routes, flow of receivers on tankers (timing and density).

The subjective evaluation forms required the participants to rate the interfaces in terms of their usefulness to specific decision phases. To ensure all participants shared a common understanding of the terms involved, there was a brief presentation on fundamental processes of decision-making. This presentation described a five-phased decision process incorporating: 1) problem identification, 2) situation assessment, 3) option generation, 4) option evaluation, and 5) option selection. Participants were told their tasks would involve all steps, except option generation.

Participants next were instructed in the terminology used by NASA-TLX to describe workload dimensions. Each workload term was defined and distinguished from the other terms. Participants then were shown how to mark the NASA-TLX form after each trial. Instruction on filling out the paired comparison form was conducted at the end of the session when the form was filled out.

The final phase of instruction focused on the content and operation of the interface to be used during the session. At the first session, participants trained on a copy of the interface equipped with one sample trial. The instructor presented each window, pointing out specific information the window provided.

Participants also learned how to select, re-size and move windows to find the information desired. Following this training, participants spent approximately 20 minutes experimenting with the interface and trial tasks on the sample trial.

After a 15-minute break, participants proceeded with the experiment. The total experimental interaction session was limited to 90 minutes, but participants set their own pace to complete the individual trials. After concluding each trial, the participants filled out the NASA-TLX workload assessment form for that trial. The time spent on the trial workload forms was not recorded as part of the decision trial time. At the end of the experimental session, the participants filled out the NASA-TLX forms for the session and completed the additional forms for subjective assessment of the interface and interaction.

The second interface session was conducted on the following day. Pre-session instruction briefly reviewed the training from the previous session on the task domain, decision-making phases, and the completion of the NASA-TLX forms. Following this review, the presentation proceeded with detailed instruction on the interface used during that session. Instruction reviewed the content of the windows common to both interfaces and reminded participants of the operation conventions involved in using the menus and moving or re-sizing windows. Participants were informed of the difference in the interface (i.e., the presence or absence of the ***Option View*** window), but no suggestions were made on how they might adjust their information search by using a new window or finding similar information in the remaining windows. The experimental session was conducted in the same manner as the first session.

At the close of the second session, after all data were collected, there was an open discussion to allow the subjects to make comments about the two interfaces, the decision tasks, and the experimental process. Subjects were also encouraged to describe their own decision processes. This discussion was audio taped to permit later analysis. The experimenter concluded by providing additional information about the goals of the study.

4.1.5. Dependent Measures

The principal hypothesis of the evaluation effort predicted the CSE system interface would improve the cognitive processes involved in decision-making tasks and, therefore, improve overall decision performance. Several objective and subjective measures were developed to assess the effect of each interface on the cognitive processes of interest. The experimental software captured the following objective data on each trial:

- Time - trial start, trial stop, and time of decision selection;
- Process - time stamped trace of screens viewed; and
- Choice - option chosen.

In addition, manual forms were used to collect the following subjective information:

- Workload - NASA TLX forms;
- Usefulness for Tasks - subjective screen evaluations; and
- Usability/Utility - free-form interface evaluations.

To permit a more realistic focus on the decision tasks, subjects were not required to complete any forms during the actual decision trials. Between-trial tasks were limited to the workload rating for the trial.

Objective Measures

Decision Performance Measures

Although TDO task performance is not typically assessed using discrete outcome measures, the length of time required to find a satisfactory solution does represent one reasonable measure of effectiveness (MOE) for replanning during ATO execution. The task aspects of this MOE were incorporated in the outcome measures developed to assess TDO decision performance.

Decision Speed

Decision speed was defined as the time in seconds to complete an individual decision trial. It was hypothesized that the CSE interface would improve the TDOs' situational understanding, focus their information search on the key variables, and reduce their workload, resulting in an overall reduction in the time required to make the decision. Time was recorded from decision maker's "acceptance" of a new task by clicking on the "Show Task" button in the ***Task Notify*** window of the current trial until the user chose an option by clicking on the "Done" button in the ***Option Select*** window. Choosing the ***Option Select*** window restricted the subject from further review of the decision trial windows.

Decision Accuracy

Decision accuracy was defined as the subject's scores in choosing the "better" replanning option from the two presented in each trial. As indicated previously, it was hypothesized that decision-oriented ***Option View*** window in the CSE interface would support better situational understanding and option evaluation leading to a larger percentage of correct decisions.

Decision Process Measures

Measures were developed to assess two factors affecting the cognitive processes involved in situational understanding and decision-making. These factors, *focus of information review* and *viewpoint selection*, provide one MOE for the decision-makers' use of information in decision-making. It was hypothesized that when the decision-maker had a better understanding of the decision situation, their information review processes would be more focused on the key decision information and their problem view would be expanded to provide an overview perspective on the operational environment. Conversely, decision-makers with a poorer understanding of the situation would tend to sift through more information hoping to discover something that would illuminate the situation with a resulting contraction of their viewpoint to more detailed displays. The process tracing capability of the data capture software provided a non-intrusive measure of these information use behaviors.

Window Changing

At the simplest level, in addition to general reduction in decision time, a more focused information search should result in fewer window changes. That is to say, decision-makers would not flip around the interface, but would focus on a few specific windows. Information reviewing behavior may then be quantified by counting the mean number of window changes in a decision trial. Thus, it was hypothesized that decision-makers using the CSE interface would exhibit less window changing activity than those using the original interface.

Problem View Selection

Achieving and maintaining adequate situational awareness and understanding in a highly complex, dynamic environment requires decision-makers to adjust their problem view to provide an adequate overview with ready access to the informational details required to make a rapid and accurate decision. The requirement for this view adjustment capability was raised several times by the operational personnel interviewed in the force-level commands and the FLEX Working Group. To examine this aspect, the windows in the two interfaces may be categorized in terms of the nature of the presentation and level of detail in each display (Table 4.2). For example, the ***Option View*** window in the CSE interface features a lower detail, graphical abstraction of tanker operations over a 24-hour period. In contrast, the ***Tanker Worksheet*** window available in both interfaces, features a high-detail, data display of air refueling mission information. This categorization of detail refers to the level of abstraction in the data presented to the user, rather than the degree of detail underlying the presentation or the resolution of the display.

Graphic Overview Windows	Alphanumeric Data Windows	Single Purpose Windows
Marquee Map Graphic Option View *	Task Inspector Tanker Worksheet Tanker SDB	Task Notify Replanning Options

Table 4.2: Classification of Interface Windows by Level of Detail Presented
(* Option View available only in CSE Interface)

It was expected that decision-makers exhibiting more focused information reviewing behavior would elect to spend more of their time in the graphic overview windows and use the alphanumeric data windows only when specifically required to retrieve relevant details. In contrast, the less focused review behavior would result in more frequent selection of the detailed data windows and longer viewing times as decision-makers read through both the relevant and irrelevant data. Thus, subjects using the CSE interface were hypothesized to have a greater percentage of graphic overview window selections than subjects using the original interface; conversely, subjects using the original interface were expected to have a greater percentage of detailed data display window selections than subjects using the CSE interface. Subjects using the CSE interface were expected to spend a larger percentage of their total decision time viewing graphic overview windows; subjects using the original interface were expected to spend a larger percentage of their total decision time viewing detailed data windows. Finally, it was hypothesized, that once a specific detailed data window was selected, subjects using the CSE interface would spend a shorter period of time viewing the window than subjects using the original interface. Again, this hypothesis was based on the assumption that CSE interface would help to focus the decision-makers' search on specific information items. These hypotheses are summarized in Table 4.3 below.

HYPOTHESIS	INTERFACE	
	Original	CSE
Larger % Window Selections	Detailed Data	Graphic Overview
Larger % Decision Time	Detailed Data	Graphic Overview
Average time viewing selected detailed data window	Longer	Shorter

Table 4.3: Summary of Problem View Hypotheses

Subjective Measures

Cognitive Workload Measures

The complexity of the decision tasks and the volume of information facing the COD decision-makers mandates an assessment of the potential cognitive workload associated with any proposed decision support system. In this sense, the MOE involved represents a measure of the decision aid's effectiveness in reducing cognitive workload rather than a measure of the decision-maker's performance. The NASA Task Load Index (TLX) was selected for its sensitivity and validity in assessing the contribution of multiple factors in the overall perception of task load (Hart & Staveland, 1988). In comparative studies of four workload measures, TLX was superior in both sensitivity and subject acceptance (Hill *et al*, 1992). After each trial, participants rated the applicable system interface on five task dimensions: mental demand, temporal demand, effort, own performance, and frustration level.[3] At the close of each session, participants evaluated the interaction session with a series of paired comparisons of the five dimensions to determine the weight (or relative rank) of the subjective importance of each dimension. The overall workload for the interaction session was calculated for each interface by multiplying the subject's weight and rating for each workload dimension and summing the products.

It was hypothesized that the cognitive workload experienced by decision-makers using the CSE interface would be less than that experienced using the original interface. Since one of the stated goals of the CSE framework is improving the support to decision-makers operating in complex, dynamic environments, it was expected that evidence of workload reduction would be statistically significant in certain dimensions. Specifically, it was hypothesized that subjects would experience less workload in the dimensions associated with complexity (i.e., mental demand) and time pressure (i.e., temporal demand) when using the CSE interface.

Usefulness and Usability Measures

Three subjective instruments were developed to capture the decision-makers' opinions regarding the usefulness of the specific interface windows to perform decision subtasks and general usability. These evaluations represented subjective MOEs for the decision aid's support to specific tasks and user preference.

Window Usefulness for Specific Decision Phases

At the end of each interaction session, participants rated the interface windows with respect to each window's contribution to the four phases of decision-making involved in the decision task: problem identification, situation assess-

---

[3] NASA TLX includes a dimensional measure for Physical Effort (PE). Since the FLEX system does not require target "hooking" or similar physical response actions, the PE dimension was excluded from the rating forms and the weighting method was adjusted to five dimensions.

ment, option evaluation, and option selection. For example, the question assessing the window's usefulness in the situation assessment task was expressed as follows:

> To what extent did this window contribute to your understanding of the situation (i.e., location & scheduled availability of resources)?
>
> **not at all** **somewhat** **greatly**
>
> 1 2 3 4 5 6 7 8 9 10 11

It was hypothesized that, whether using the original or the CSE interface, decision-makers would rate the graphic overview windows higher than the detailed data display windows for situation assessment, option evaluation, and option selection tasks. It was further hypothesized that the ***Option View*** window in the CSE interface would be rated higher than the other windows on these tasks.

Since the ***Marquee*** (available in both interfaces) was the closest analog to the ***Option View*** window in the CSE interface, hypotheses were developed about the potential changes in the use of the ***Marquee*** based on the availability of the ***Option View*** window. The ***Marquee*** and ***Option View*** windows both present graphic overviews. The ***Marquee,*** while it is an extraordinarily flexible interface to the detailed data, does not present all the critical decision variables. For example, decision-makers must look elsewhere to determine how much taskable fuel would remain if a particular option were chosen. Furthermore, there is no way to visually compare options at the ***Marquee***. Finally, despite the mission flow features of the ***Marquee,*** it is more difficult to visualize the impacts of a particular choice on the entire tanker operations mix. For these reasons, it was hypothesized that the ***Option View*** window would replace the ***Marquee*** in the user's preference when available. Conversely, when decision-makers used the original interface after using the CSE interface, it was hypothesized that they would shift preference to the ***Marquee***.

Free-Form Subjective Interface Evaluations

Participants were allowed two opportunities to express their general opinions and preferences regarding the two interfaces. After each interaction session, subjects were asked to provide answers to the following questions:

1. What did you like and/or find most helpful about this interface?
2. What did you dislike and/or find most difficult about this interface?

In addition, subjects participated in an open discussion of both interfaces after the final interaction session. Both the questionnaire and the discussions were intended to provide explanatory information to aid in the interpretation of the data collected in the structured formats. Nevertheless, it was expected that participants would express favorable opinions regarding the CSE interface in general and the ***Option View*** window in specific.

4.1.6. Results

The majority of the hypotheses were supported at statistically significant levels. Decision-makers using the CSE interface performed decision tasks faster and more accurately while experiencing less workload due to mental and temporal demands. This section presents the results of experimental study. The results for the objective measures of decision performance and process are presented first, followed by the subjective measures of cognitive workload, interface usefulness, and user preference.

Objective Measures

Results for most of the objective measures were statistically significant with $p < .0001$. This section presents the results for objective measures of decision performance (task speed and accuracy) followed by those for decision process (window changing and problem view selection).

Decision Performance Measures

A 2 (interface) x 2 (order) ANOVA was performed for both objective performance measures: decision task speed and accuracy. The system interface employed was a within-subject variable; the order the subject was exposed to the two system interfaces was a between-subject variable. Decision speed was defined as the average time in seconds to complete an individual decision trial. Decisions were scored for accuracy based on a match to the "better" choice (determined by expert consensus) and reported as the percentage of correct answers. Table 4.4 presents the results of the ANOVAs for both main effects (interface and exposure order) and interaction (interface x order) for each of the objective performance measures. The interface main effect was significant for both task speed and decision accuracy. The order main effect was significant for

Measure	Interface Main Effect	Order Main Effect	Interface x Order Interaction
Task Speed (seconds)	p < .0001 MSE = 198240.06 F(1,10) = 654.33 Original = 203.69 CSE = 151.22	p < .05 MSE = 334971 F(1,10) = 6.85 Orig-1st= 211.57 CSE-1st = 143.35	p < .0001 MSE = 550200 F(1,10) = 1816.05 Orig-1st CSE-1st Orig 281.50 125.88 CSE 141.61 160.82
Decision Accuracy (% Correct)	p < .0001 MSE = 1.125 F(1,10) = 376.170 Original = 71.53 CSE = 84.03	NS MSE = 0.014 F(1,10) = 0.110 Orig-1st= 77.08 CSE-1st = 78.49	p < .0001 MSE = 0.2222 F(1,10) = 74.305 Orig-1st CSE-1st Orig 68.06 75.00 CSE 86.11 81.94

Table 4.4: ANOVA Results: Decision Performance Measures

task speed only. The interaction of interface and order was significant for both measures.

Task Speed

Figure 4.3a graphically depicts the significant improvement in average decision time (translated from seconds to minutes) achieved with the CSE Interface: F(1,10) = 654.33, MSe = 198240.06, p < .0001, original interface mean = 203.69 secs (3.39 min), CSE interface mean = 151.22 secs (2.52 min). The CSE interface resulted in an average decision time of almost 1 minute, approximately 25% less than the average for the original interface.

The main effect for exposure order was also significant. As depicted in Figure 4.3b, decision-makers who used the interfaces in the CSE-1st order (CSE interface followed by original interface) performed the decision tasks with an average time that was significantly faster than that of the decision-makers who used the interfaces in the Orig-1st order (original interface followed by CSE interface): F(1,10) = 6.85, MSE = 334971, p < .05, Orig-1st order mean = 211.57 secs (3.53 min), 143.35 secs (2.39 min). [4]

Examination of the interface by order interaction further illuminates the results of the main effects. As depicted in Figure 4.3c, the interaction of the interface used and exposure order also had a significant effect on decision speed: F(1,10) = 1816.05, MSE = 550200, p < .0001. The most dramatic effect was the change in average decision time while using the original interface based upon the exposure order. When the CSE interface was used first (exposure order = CSE-1st), average decision time for decision-makers using the original interface was reduced by more than 2-1/2 minutes. In fact, the fastest average decision time was achieved with the CSE-1st exposure order. In contrast, the difference in decision speed based on exposure order for the CSE interface was minimal. This

---

[4] In the Task Speed graphs, the smaller number is faster, therefore, better.

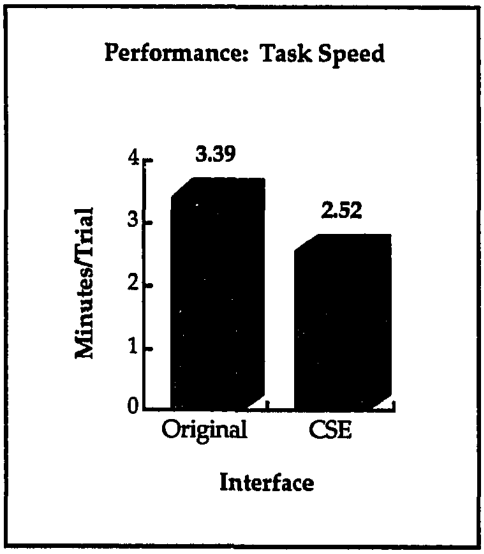


Figure 4.3a: Speed - Interface Effect (p < .0001)

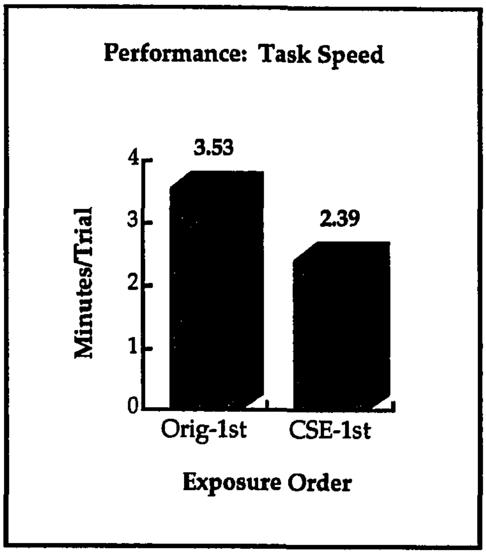


Figure 4.3b: Speed - Order Effect (p < .05)

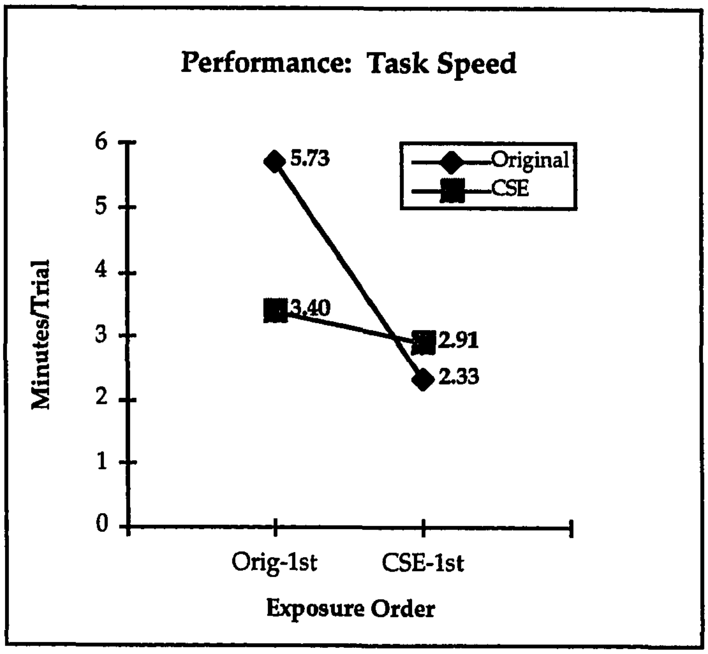


Figure 4.3c: Speed - Interface x Exposure Order Interaction (p < .0001)

difference (approximately 19 secs) is not significant and may be the result of task or interface learning effects. The impact of the interface by exposure order interaction for the decision performance measures is discussed further after the decision accuracy results are presented.

Decision Accuracy

Figure 4.4a graphically depicts the significant improvement in average decision scores (represented as the percentage of correct answers) achieved with the CSE Interface: F(1,10) = 376.17, MSE = 1.13, $p < .0001$, original interface mean = 71.54%, CSE interface mean = 84.03%. The CSE interface resulted in an average decision score approximately 12.5% better than the average for the original interface.[5]

The main effect for exposure order was not significant. As depicted in Figure 4.4b, decision-makers who used the interfaces in the CSE-1st order performed the decision tasks with an average score that was slightly higher than that of the decision-makers who used the interfaces in the Orig-1st order.

As depicted in Figure 4.4c, the interaction of the interface used and exposure order also had a significant effect on decision accuracy: F(1,10) = 74.31, MSE = 0.22, $p < .0001$. As depicted in the graph, the more dramatic effect is the change in average percentage of correct choices while using the original interface. When the CSE interface was used first (exposure order = CSE-1st), average score for decision-makers using the original interface was improved by almost 7%. Unlike the results for task speed, decision accuracy is highest for the CSE interface regardless of exposure order.

---

[5] In the Decision Accuracy graphs, the larger number represents the higher percentage correct and is, therefore, better.

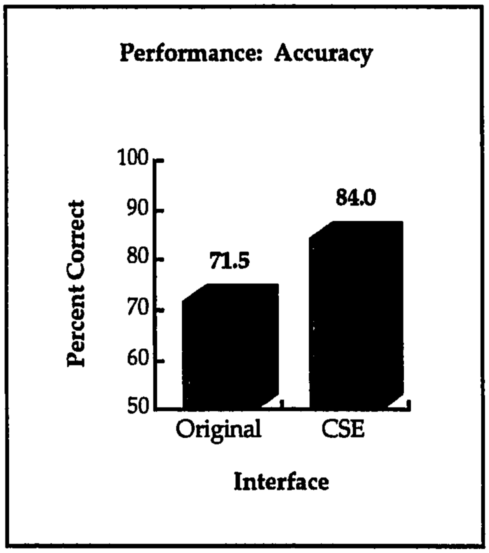


Figure 4.4a: Accuracy - Interface Effect (p < .0001)

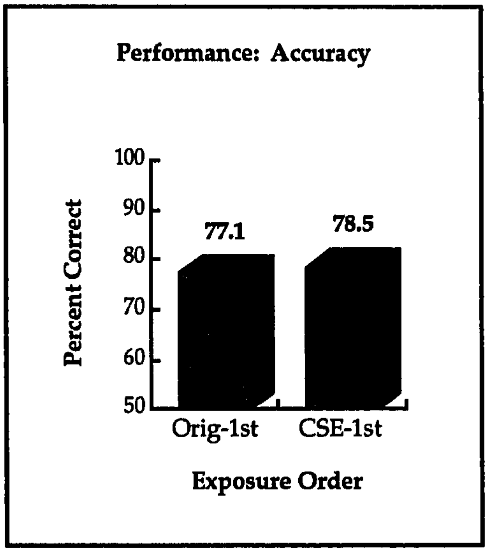


Figure 4.4b: Accuracy - Order Effect (not significant)

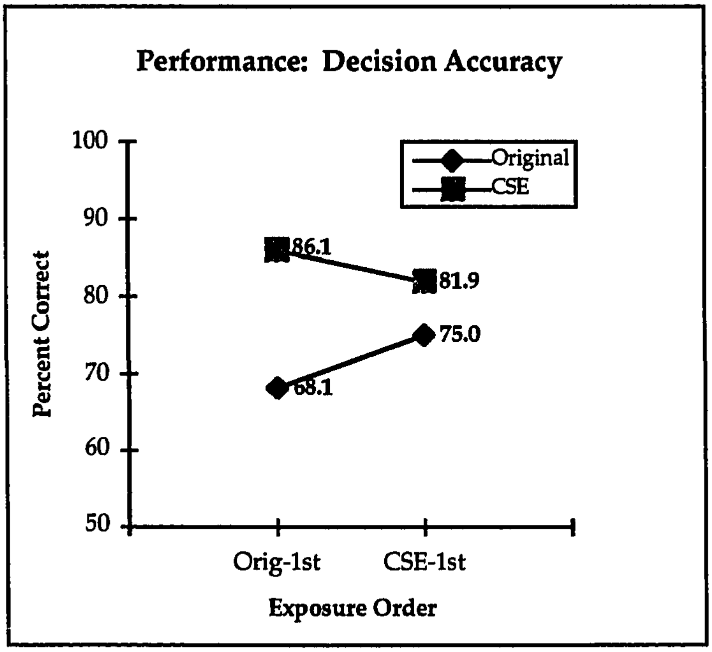


Figure 4.4c: Accuracy - Interface x Exposure Order Interaction (p < .0001)

The Interface by Exposure Order Interaction

As reported above, results of the ANOVAs for the interface by exposure order interaction were significant for both objective performance measures. In both cases, exposure order had little effect on the performance of decision-makers using the CSE interface. These results, coupled with the significant interface main effects favoring the CSE interface, suggest that the CSE interface may afford the user the ability to get "up to speed" faster with the task. This, in part, may be attributable to the CSE interface's system-level model (presented in the ***Option View*** window) assisting the decision-maker in handling the complexities of the task by engendering a useful mental model of the operational interdependencies. The possible presence of such a model is consistent with the effects of exposure order on the performance of decision-makers using the original interface. Decision performance using the original interface was significantly better when the users were first exposed to the CSE interface. In the open discussions, conducted after both interaction sessions were completed, several participants in the CSE-1st exposure groups reported that they retained the overview model when the ***Option View*** window was unavailable (original interface condition). They used this "mental" model with the ***Marquee*** window and detailed information in the ***Tanker Worksheet*** and ***Tanker SDB*** windows. Although no measures were developed to examine the presence or content of the user's mental models of the task and domain, the decision process measures do seem to support this interpretation of the interaction results.

Decision Process Measures

A 2 (interface) x 2 (order) ANOVA was performed for each of the objective decision process measures. As with the performance measures, the system interface employed was a within-subject variable; the order the subject was exposed to the two system interfaces was a between-subject variable. The measures analyzed for decision process included:

- Window Changing - the average number of window changes per decision trial;
- Problem View Selection - examined the use of various information presentation formats, including
    - » percentage of graphic overview window selections versus detailed data window selections;
    - » percentage of time spent in graphic overview versus detailed data windows; and
    - » average time spent in detailed data windows once selected.

The results of the ANOVAs for all the process measures were significant, with the exception of the average time spent in detailed data windows. The results of each analysis are presented below.

Window Changing

Window changing was measured as the mean number of window changes per decision trial. Table 4.5 presents the results of the ANOVAs for the main effects (interface and exposure order) and interaction (interface x order) for the window changing measure. Results for both main effects were significant, as were the results for the interface by exposure order interaction.

Since the key decision information was integrated into the ***Option View*** window in the CSE interface, but scattered across several windows in the original interface, logic suggested that subjects using the CSE interface would focus on the ***Option View*** window and exhibit less window changing activity. In contrast, the original interface offered no simple way to avoid "hunting" for the necessary information. The window changing hypothesis depended upon the decision-maker recognizing that the key information was present in the ***Option View***

Measure	**Interface Main Effect**	**Order Main Effect**	**Interface x Order Interaction**
Window Changes (#/Trial)	p < .0001 MSE = 1790.17 F(1,10) = 78.84 Original = 10.88 CSE = 7.78	p < .05 MSE = 7905.12 F(1,10) = 8.52 Orig-1st = 11.31 CSE-1st = 7.25	p < .0001 MSE = 7114.75 F(1,10) = 313.33 Orig-1st CSE-1st Orig 13.43 6.24 CSE 7.40 8.11

Table 4.5: ANOVA Results for Decision Process Measures - Window Changing

window. The pre-test training only informed the user as to the information found in each window, without emphasizing the value of the information. Consistent with the hypothesis, decision-makers changed windows less (i.e., did not "wander" around the interface as much) when using the CSE interface than when using the original interface: F(1,10) = 78.84, MSE = 1790.17, p < .0001. As graphically displayed in Figure 4.5a, the average number of window changes per trial was 10.9 for the original interface versus 7.8 changes per trial for the CSE interface.

The results of the ANOVAs for the exposure order main effect were also significant: F(1,10) = 8.52, MSE = 7905.12, p < .05. As reported with the objective performance measures, the CSE-1st interface exposure order produced the better result. As indicated in Figure 4.5b, the Orig-1st exposure order resulted in an average of 11.3 window changes per trial; the CSE-1st exposure order averaged only 7.2 window changes per trial.

The ANOVA results for the interface by exposure order interaction were also significant: F(1,10) = 313.33, MSE = 7114.75, p < .0001. As discovered previously with the objective performance measures, this analysis indicated that using the CSE interface first had a significant effect on the subsequent use of the

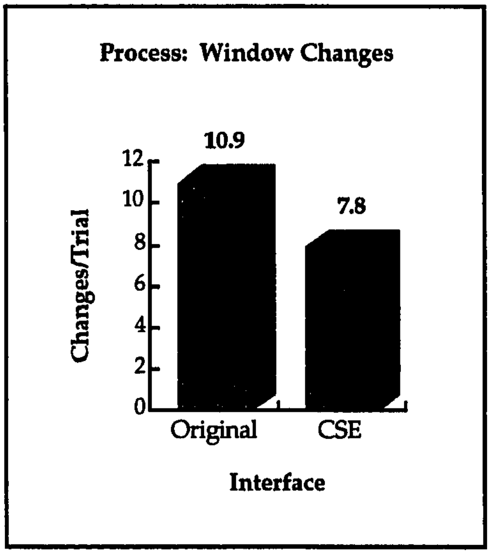


Figure 4.5a: Average Window Changes per Trial - Interface Effect (p < .0001)

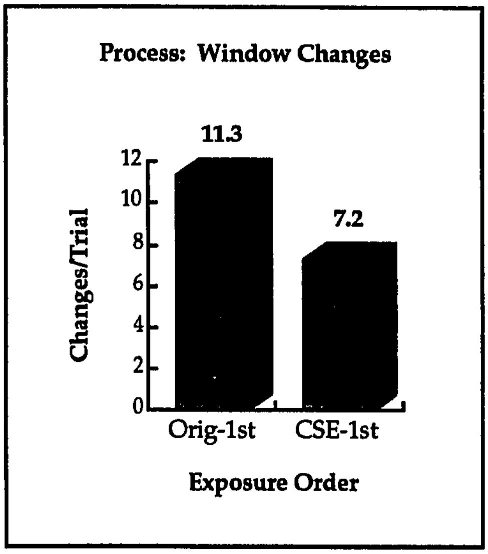


Figure 4.5b: Average Window Changes per Trial - Order Effect (p < .05)

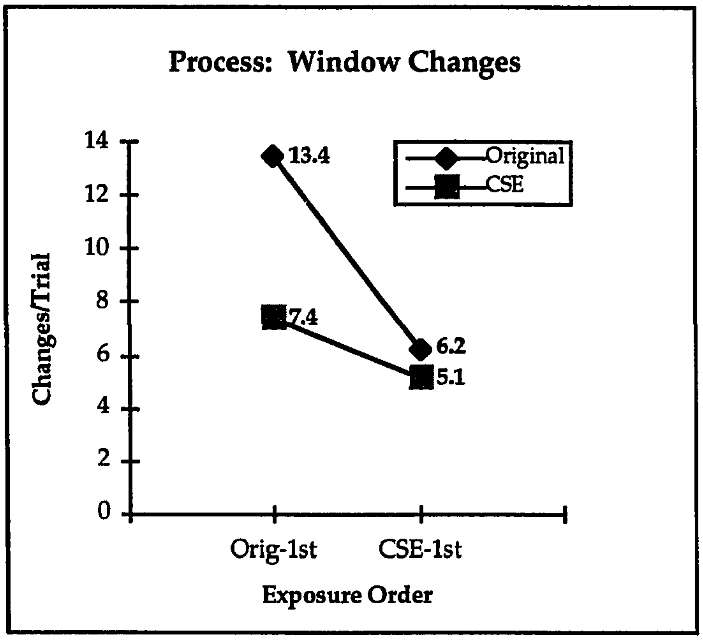


Figure 4.5c: Average Window Changing per Trial - Interface x Exposure Order Interaction (p < .0001)

original interface. As depicted in Figure 4.5c, when decision-makers used the CSE interface first, the average number of window changes per trial was only 5.1 (the smallest average recorded). Subsequent use of the original interface resulted in an average of 6.2 changes per trial (the next smallest average recorded). In contrast, when using the original interface first, decision-makers changed windows an average of 13.4 time per trial; subsequent use of the CSE interface resulted in an average of 7.4 window changes per trial.

The results of the CSE-1st exposure order interaction with the two interfaces is similar to that reported for the performance measures. This suggests the possibility of a general trend supporting the hypothesis that the users gained some particular advantage when using the CSE interface first, perhaps in the form of a retained mental model, that improved subsequent performance with the original interface. In contrast, the results of the Orig-1st exposure order interaction with the interfaces suggests that using the original interface first may have *negatively* affected the subsequent use of the CSE interface. Intuitively, the second interaction session should always result in better performance due to the additional experience with the task and the common windows of the interface. Despite this second session advantage, results indicate that decision-makers exhibited a more focused information review when the CSE interface was used first as opposed to when the CSE interface was used after the original interface. Data were not collected to fully explore this issue; however, some additional insight is provided by the examination in the next section of how decision-makers used the two interfaces.

Problem View Selection

As discussed previously, the original interface and the CSE interface shared seven common windows. Of these six, three windows (***Task Inspector, Tanker Worksheet***, and ***Tanker SDB***) presented information as highly-detailed alphanumeric data; two windows (***Marquee*** and ***Map Graphic***) presented information in a graphical overview format; the remaining windows (***Task Notify*** and ***Replanning Options***) presented single pieces of information and were excluded from the analysis. The ***Option View*** window, available only in the CSE interface, also presented information in the graphic overview format. The analyses in this section examine the decision-maker's use of the two principal display formats with respect to (1) the percentage of window changes selecting each category, (2) percentage of decision time spent in each category, and (3) the average time spent in a detailed window after selection. The associated hypotheses were summarized previously in Table 4.3. As in previous analyses, 2 (interface) by 2 (exposure order) ANOVAs were performed to evaluate each hypothesis.

Window Usage - Selection of Information Presentation Format

The first analysis performed examined the window changing activity to determine the percentage of window selections associated with each of the two categories. Table 4.6 presents the results of the ANOVAs for display format selection. Consistent with the hypotheses, examination of the interface main effect indicated that decision-makers selected graphic overview windows more often when using the CSE interface (60.7%) versus the original interface (42.6%): F(1,10) = 619.47, MSE = 11.03, p < .0001 (Figure 4.6a). Conversely, decision-makers selected detailed data windows more often when using the original interface (34.9%) versus the CSE interface (25.4%): F(1,10) = 147.05, MSE = 2.11, p < .0001.

Results of the ANOVAs for the exposure order main effects were also significant (see Table 4.6). As presented in Figure 4.6b, the graphic overview windows were selected more often in the CSE-1st exposure order, while the detailed data windows were selected more often in the Orig-1st exposure order.

Measure	Interface Main Effect	Order Main Effect	Interface x Order Interaction
Graphic Overview Selection (% of changes)	p < .0001 MSE = 11.03 F(1,10) = 619.47 Original = 42.6 CSE = 60.7	p < .01 MSE = 7.51 F(1,10) = 18.27 Orig-1st = 45.0 CSE-1st = 57.5	p < .0001 MSE = 1.13 F(1,10) = 63.19 Orig-1st CSE-1st Orig 37.54 51.79 CSE 58.76 62.39
Detailed Data Selection (% of changes)	p < .0001 MSE = 2.11 F(1,10) = 147.05 Original = 34.9 CSE = 25.4	p < .01 MSE = 8.54 F(1,1) = 21.39 Orig-1st = 36.6 CSE-1st = 23.2	p < .0001 MSE = 2.24 F(1,10) = 156.17 Orig-1st CSE-1st Orig 41.32 23.21 CSE 27.84 23.26

Table 4.6: ANOVA Results for Decision Process Measures - Information Presentation Format Selection

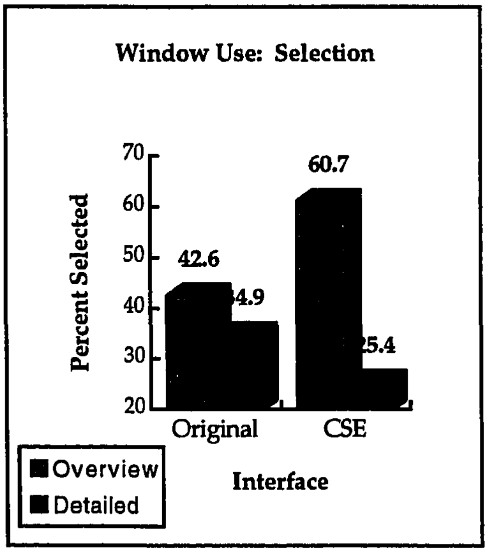


Figure 4.6a: Presentation Format Selection - Interface Effect (p < .0001)

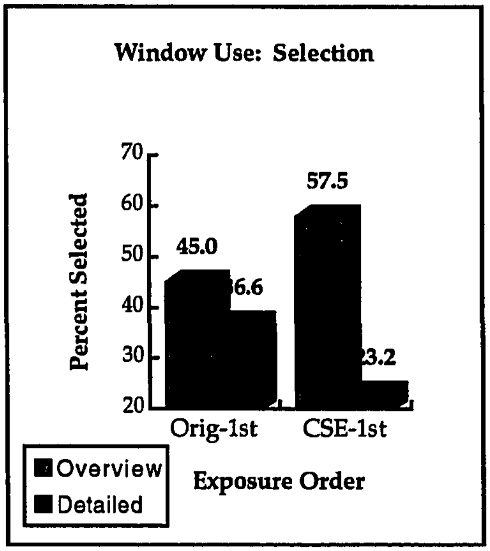


Figure 4.6b: Presentation Format Selection - Order Effect (p < .01)

The results for the ANOVAs for the interface by exposure order interaction were significant for both the use of the graphical overview windows and the detailed data windows (see Table 4.6). As seen in the analysis of the objective performance measures, the most dramatic interaction was that of the exposure order difference in the view selection of decision-makers using the original interface. As indicated in Figure 4.6c, decision-makers using the original interface substantially increased their selection of the graphic overview windows in the groups using the CSE interface first (CSE-1st exposure order). The mean percentage of window changes selecting graphic overview windows in the original interface condition and the CSE-1st exposure order was 51.8% versus a mean of 37.5% in the Orig-1st exposure order. In contrast, decision-makers using

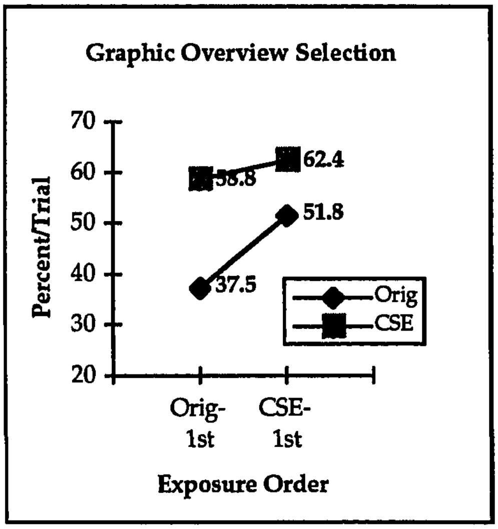


Figure 4.6c: Presentation Format Selection - Interface x Exposure Order Interaction (p < .0001)

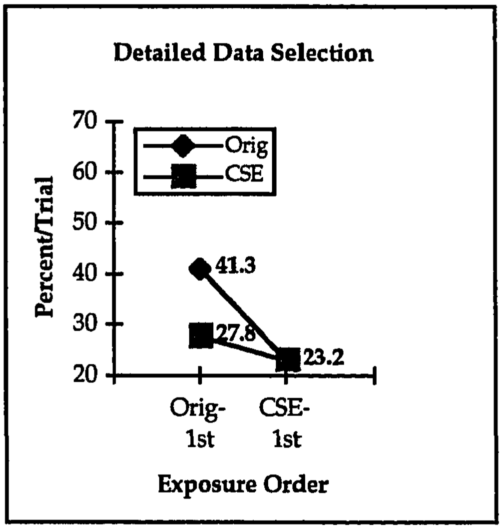


Figure 4.6d: Presentation Format Selection - Interface x Exposure Order Interaction (p < .0001)

the CSE interface varied less than 6% in their selection of graphic overview windows (mean Orig-1st order = 56.8% versus mean CSE-1st order = 62.4%).

Results of the ANOVAs to analyze the interface by exposure order interactions for the percentage of detailed data window selections were significant (see Table 4.6). As hypothesized, the direction of the change in selection was the reverse of that for the graphic overview windows. Decision-makers using the original interface exhibited the largest percentage (mean = 41.3%) of detailed data window selections in the Orig-1st exposure order (Figure 4.6d). In the CSE-1st exposure order condition, decision-makers using the original interface were within .05% of their selection percentage while using the CSE interface (CSE-1st order: mean original = 23.26%; mean CSE = 23.21%). There was less than a 3% difference in the two exposure orders for users of the CSE interface. The interaction of interface and exposure order is discussed further in the section on the percentage of time spent.

Window Usage - Percentage of Time Spent in Presentation Format Types

In addition to the percentage of window selections, the percentage of time spent in the two presentation format categories was analyzed to control for the possibility of windows being selected in error (i.e., slips and mistakes in interface control). Thus, consistency between percentage of selections and percentage of time spent should strengthen the implications of both sets of findings. The results of the ANOVAs for percentage of time spent in each category were significant for both main effects and the interface by exposure order interaction (Table 4.7).

Figure 4.7a graphically presents the interface main effects for the percentage of time spent in both the graphic overview and the detailed data windows. Consistent with the hypotheses summarized earlier in Table 4.3, the percentage of time spent in the graphic overview window was greater when the decision-makers used the CSE interface (mean original = 41.7% versus mean CSE = 59.8%). Conversely, percentage of time spent in detailed data windows was greater when decision-makers used the original interface (mean original = 37.2% versus mean CSE 27.3%). The difference between graphic overview and detailed data window usage was greatest in the CSE interface condition (graphic overview mean = 59.8%; detailed data mean = 27.3%).

Measure	Interface Main Effect	Order Main Effect	Interface x Order Interaction
Graphic Overview Usage (% of time)	p < .0001 MSE = 10.72 F(1,10) = 387.92 Original = 41.7 CSE = 59.8	p < .05 MSE = 4.74 F(1,10) = 5.136 Orig-1st = 45.2 CSE-1st = 55.1	p < .0001 MSE = 1.85 F(1,10) = 66.83 Orig-1st CSE-1st Orig 37.0 50.3 CSE 60.3 59.3
Detailed Data Usage (% of time)	p < .0001 MSE = 2.48 F(1,10) = 108.64 Original = 37.2 CSE = 27.3	p < .01 MSE = 5.96 F(1,10) = 15.88 Orig-1st = 37.8 CSE-1st = 26.6	p < .0001 MSE = 4.22 F(1,10) = 185.11 Orig-1st CSE-1st Orig 43.6 25.5 CSE 27.0 27.5

Table 4.7: ANOVA Results for Decision Process Measures - Percentage of Time Spent in Different Presentation Formats

The exposure order main effects were significant for the average time spent in both formats (see Table 4.7). Figure 4.7b graphs the mean time spent in each category given the order of interface exposure. As in previous analyses, this difference was most pronounced in the CSE-1st exposure order. Decision-makers in these groups averaged 55.1% of their decision time in the graphic overview windows versus 26.6% in the detailed data windows for a difference of almost

29%. Decision-makers in the Orig-1st order groups averaged 45.2% in the graphic overview windows versus 37.8% in the detailed data windows for a difference of only 7.4%. As in the case of the interface main effects, the results for the exposure order main effects appear to support the results of the analysis of the percentage of window selections.

The ANOVA results for the interaction of interface and exposure order provide additional insight into the window use patterns (Table 4.7). Figures 4.7c-d graph the interface by exposure order interaction for both information presentation formats. Again, as with the results for the main effects, the interaction results support the previous findings for percentage of window selections. CSE interface users displayed a very small change due to exposure order (average percent difference = 1% for time spent in graphic overview windows and 0.5% for time spent in detailed data windows). In contrast, when decision-makers used the original interface, exposure order made a significant difference. Time spent in graphic overview windows varied from 37.0% in the Orig-1st exposure order to 50.3% in the CSE-1st exposure order. Similarly, time spent in the detailed data windows varied from 43.6% in the Orig-1st exposure order to 25.5% in the CSE-1st order. In the CSE-1st exposure order, the percentage of time spent using the original interface was within 2% of the CSE interface condition for both exposure orders.

The interaction findings for both the percentage of window selections and percentage of time spent appear to support those discussed previously for the objective performance measures (decision speed and accuracy). When decision-makers were exposed first to the CSE interface, their performance using the original interface was considerably different than when the original interface was used first. Moreover, the change in performance or process using the original interface in the CSE-1st exposure order uniformly moved the measure of interest towards the means of the CSE interface users. This trend suggests that exposure

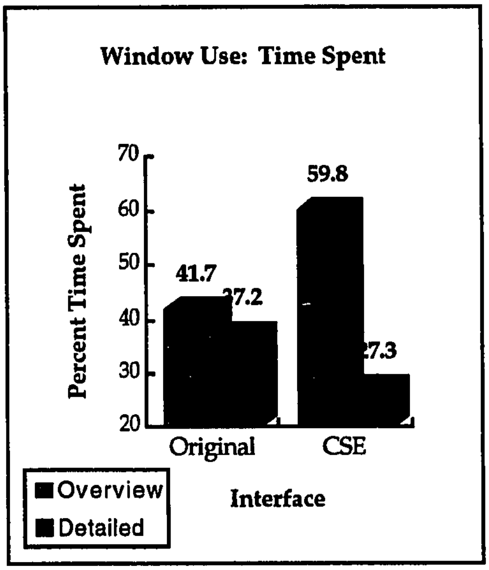


Figure 4.7a: Percentage of Time Spent in Different Formats - Interface Effect

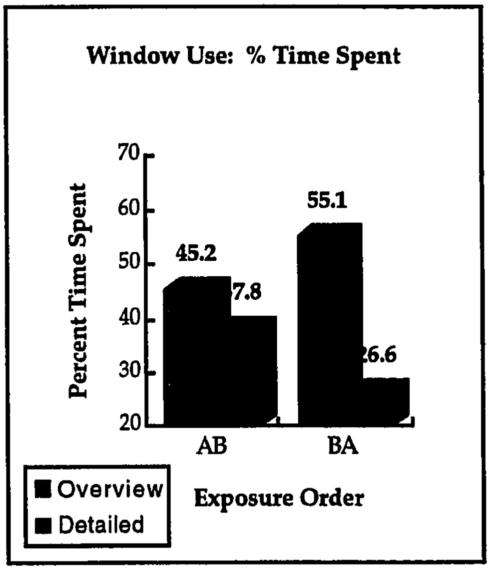


Figure 4.7b: Percentage of Time Spent in Different Formats - Order Effect

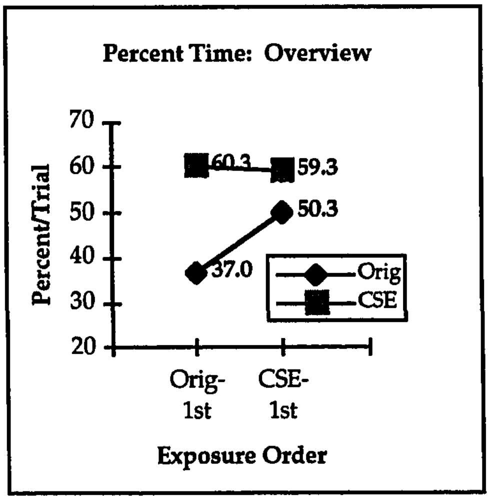


Figure 4.7c: Percentage of Time Spent in Graphic Overview Formats - Interface x Exposure Order Interaction (p < .0001)

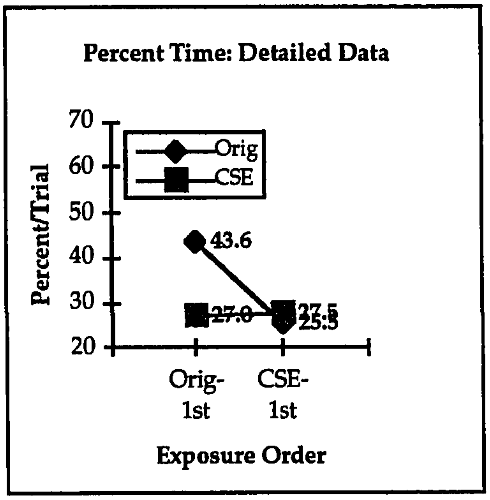


Figure 4.7d: Percentage of Time Spent in Detailed Data Formats - Interface x Exposure Order Interaction (p < .0001)

to the CSE interface first provided some manner of retained mental model or learning effect that helped users of the original interface to achieve the more desirable results possible with the CSE interface.

Window Usage - Average Time Spent in Detailed Data Windows

The third hypothesis regarding the effects of the interface on decision processes further probed the use of detailed data windows. It was hypothesized initially that, when the decision-makers selected a detailed data window, they would spend more time reviewing data in that window if they were using the original interface. The results of the ANOVAs for the interface and exposure order main effects were not statistically significant. As indicated by the average times in Table 4.8, there was a slight difference (approximately 2 seconds) in the direction of the hypothesis.

The results of the ANOVA for the interface by exposure order interaction were significant: $F(1,10) = 929.19$, $MSE = 4925.61$, $p < .0001$ (Table 4.8). As indicated in Figure 4.8, exposure order had little effect (a difference of 0.3 seconds) on decision-makers using the CSE interface. As seen previously, significance lay in the effect of the exposure order on users of the original interface. When decision-makers using the original interface were first exposed to the CSE interface, their viewing time in a detailed data window averaged only 20.3 seconds. This time was substantially lower than the average times for the CSE interface users in both exposure orders. The data collected does not provide explanations for this lower time. Although it is not clear that this is necessarily a desirable effect, the shorter viewing time for the original interface in the CSE-1st exposure order was associated with an overall decrease in the time required to complete a decision trial and increased accuracy (see Table 4.4). Additional research could help determine the underlying cognitive activities associated with the dramatic drop in viewing time.

Measure	Interface Main Effect	Order Main Effect	Interface x Order Interaction
Detailed Data Usage (avg. time)	NS Original = 25.03 CSE = 23.04	NS Orig-1st = 24.65 CSE-1st = 23.66	p < .0001 MSE = 4925.61 F(1,10) = 929.19 Orig-1st CSE-1st Orig 33.25 20.31 CSE 27.09 27.38

Table 4.8: ANOVA Results for Decision Process Measures - Average Time Spent in Detailed Data Windows

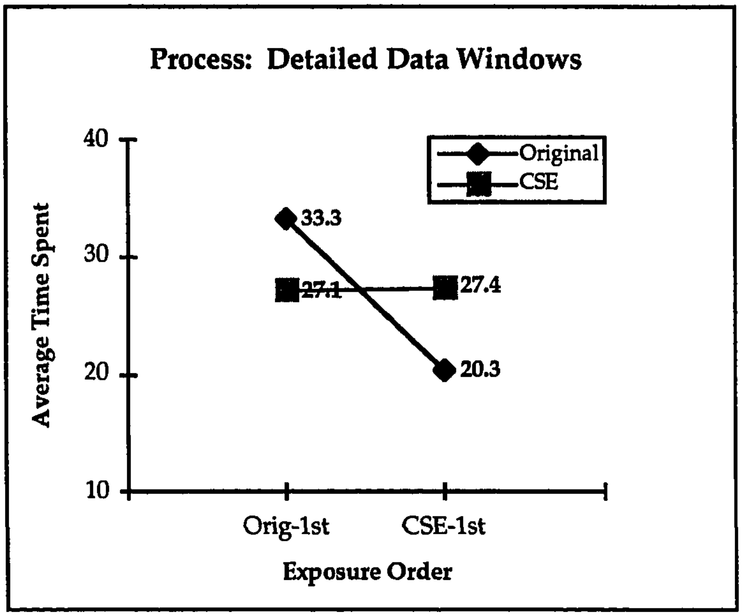


Figure 4.8: Average Time Spent in Detailed Data Windows - Interface x Exposure Order Interaction (p < .0001)

Subjective Measures

This section presents the results of the subjective evaluations. The subjective measures were designed to estimate the interface impacts on decision processes and collect decision-maker input regarding the interfaces' processes for and performance in supporting the decision-making tasks. The NASA Task Load Index (TLX) provided a subjective measure of the decision-makers' perception of the source and extent of workload associated with using each interface. Interface window evaluations collected the decision-makers' opinions on the usefulness of the specific windows in the interface for the tanker replanning decision tasks. In addition, the window questionnaires provided insight into the decision-makers' display viewpoint preferences. The free-form evaluations and open discussions supplemented and aided in the interpretation of the task load and usefulness evaluations.

Cognitive Workload Measures

The tanker replanning decision tasks principally involved task workload in the cognitive dimensions. Actual operation of the interface was restricted to using a mouse input device to select windows and indicate choices. Although there were no real-time reaction tasks, such as target "hooking" or weapons launching, the decision-maker's did participate in a moderately time-stressed condition. The NASA-TLX workload measurement instruments were used to capture the decision-makers perception of the workload they experienced while using the two interfaces.

NASA TLX

After each trial, subjects rated the applicable system interface on five dimensions: mental demand (MD), temporal demand (TD), own performance (OP), frustration level (F), and effort (E). At the close of each session, participants evaluated the interaction session with a series of paired comparisons of the five dimensions to determine the weight (or subjective importance) for each dimension. The overall workload for the interaction session was calculated for each interface by multiplying the subject's weight and rating for each workload dimension and summing the products.

First, a 2 (interface) by 2 (exposure order) ANOVA was performed on rated total workload. Interface was a within-subjects variable as all participants completed the NASA-TLX for each system interface after each trial. Exposure order was a between-subjects variable for order of the participants' interaction with the two interfaces. Three 2 (interface) x 5 (dimension) ANOVAs were performed: one each for the dimension weights, the unweighted mean scores, and the composite (weighted) mean scores. Both interface and dimension were within-subject variables; all subjects completed forms for both interfaces on all dimensions. The results of each ANOVA are discussed below.

Total Workload

It was hypothesized that decision-makers would experience less workload overall with the CSE interface. The results of the ANOVA for the interface main effect supported this hypothesis (Table 4.9). The summed weighed means for the original interface (48.89) and the CSE interface (39.18) were significantly different: $F(1,10) = 48.46$, $MSE = 6788.28$, $p < .0001$ (Figure 4.9a). There was no order main effect expected. The ANOVA results for the order main effect did show a small, but not statistically significant effect. The initial hypotheses for cognitive workload did not consider the interface by order interactions; however, the results of the objective measure analyses suggested that there might be a similar interaction effect in the workload measures. The ANOVA for the interface by exposure order interaction did produce significant results for this effect: $F(1,10) = 5.12$, $MSE = 717.78$, $p = .0244$. Figure 4.9b graphs the combined effects of interface and exposure order on the total workload.

Measure	Interface Main Effect	Order Main Effect	Interface x Order Interaction
Total Workload (Summed Weighted Means)	p < .0001 MSE = 6788.28 F(1,10) = 48.46 Original = 48.89 CSE = 39.18	NS Orig-1st = 44.58 CSE-1st = 43.50	p = .0244 MSE = 717.78 F(1,10) = 5.13 Orig-1st CSE-1st Orig 51.01 46.78 CSE 38.14 40.23

Table 4.9: ANOVA Results for NASA TLX Total Workload Analysis

As in the analyses of the interaction effects for the objective measures, the order of exposure most affected the workload experienced with the original interface. The retained mental model interpretation suggested by the objective measure interactions also may have some validity for the workload measures. When the original interface is used first, decision-makers reported the highest levels of workload. When the CSE interface is used first, subsequent use of the original interface results in a lower workload score than with the Orig-1st exposure order, but still higher than the score for the CSE interface in either exposure order. There is a small difference between the two orders in the workload scores for decision-makers using the CSE interface. The lower workload scores associated with the second interaction session for both interfaces may be due to changes in the decision-makers' general level of comfort with the tasks and interface. The next section examines the effects of the interface with respect to the individual workload dimensions.

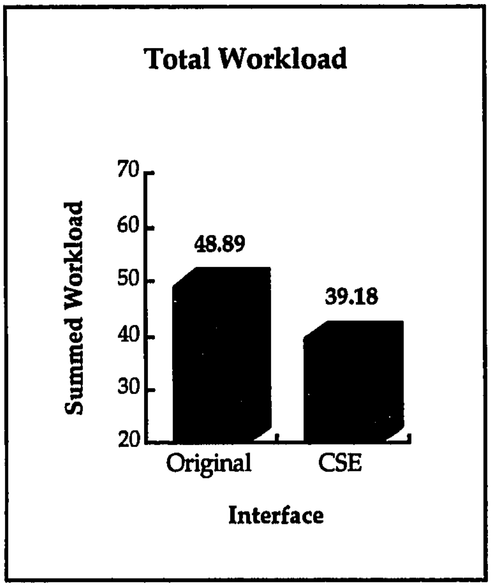


Figure 4.9a: NASA TLX - Total Workload: Interface Main Effect (p < .0001)

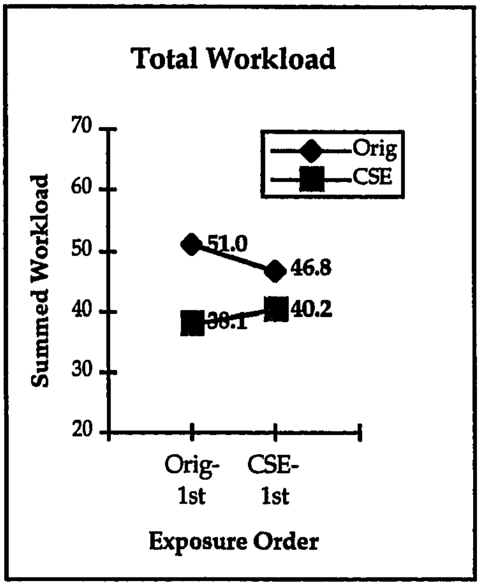


Figure 4.9b: NASA TLX - Total Workload: Interface x Order Interaction (p = .0002)

Interface Impacts on Workload Dimensions

One of the strengths of the NASA Task Load Index is its ability to measure workload in terms of multiple contributing factors rather than only providing a single measure of workload. The multi-dimensional scores of TLX are derived through the collection of workload ratings for each dimension after each trial and the paired comparisons at the end of the interaction session. This section presents the results of the ANOVAs for the TLX weights, the mean (unweighted) scores, and the composite scores for each dimension (Table 4.10).

Workload Weighting

It was hypothesized that there would be a significant main effect for dimensions in the dependent variable for the TLX weights. For example, due to the complexity of the task, it seemed probable that Mental Demand (MD) would receive higher weights than Effort (E). The interface main effect and the interface by dimensions interaction were not expected to be significant.

Results of the TLX weights ANOVA were significant for the dimensions main effect: F(4,110) = 7.24, MSE = 1.12, p < .0001. Consistent with the hypothesis, the interface main effect and the interface by dimension interaction were not significant. The cell means in Table 4.10 for the dimension main effect and the interface by dimension interaction indicate the high weighting given to Mental Demand (MD). This pattern is consistent with the hypothesis that MD would dominate due to the complexity of the decision task and environment.

Workload Means

The analysis of the unweighted workload rating means produced statistically significant results for both the interface and dimension main effects but not the interface by dimension interaction (Table 4.10). As expected, the original interface had the higher average unweighted workload score (47.56 vs. 37.81): F(1,110) = 9.14, MSE = 712.52, p = .0031. Dimension main effects yielded some surprising results. The unweighted mean scores placed Own Performance highest (61.70), followed by Effort (42.16), Mental Demand (40.98), Temporal Demand (35.57) and Frustration (33.00): F(1,110) = 9.79, MSE = 3054.22, p < .0001. The interaction means also rated OP highest for both interfaces, but the other dimensions were not matched in rank. For example, Mental Demand was ranked second for the original interface (47.64), but third highest in the CSE interface condition (36.23).

Measure	Interface Main Effect	Dimension Main Effect	Interface x Dimension Interaction
Dimension Weights	NS Original = 0.20 CSE = 0.20	p < .0001 MSE = 0.12 F(4,110) = 7.24 MD = 0.30 TD = 0.23 OP = 0.17 F = 0.13 E = 0.15	NS Orig CSE MD = 0.31 0.30 TD = 0.26 0.20 OP = 0.14 0.20 F = 0.13 0.13 E = 0.16 0.15
Dimension Mean Scores (Unweighted)	p = .0031 MSE = 712.52 F(1,110) = 9.13 Original = 47.56 CSE = 37.81	p < .0001 MSE = 3054.22 F(4,110) = 9.79 MD = 40.98 TD = 35.57 OP = 61.70 F = 33.00 E = 42.16	NS Orig CSE MD = 47.64 34.32 TD = 42.47 28.67 OP = 59.98 63.41 F = 39.60 26.40 E = 48.09 36.23
Dimension Composite Scores (Weighted Means)	NS Original = 9.779 CSE = 7.837	p = .0046 MSE = 233.679 F(4,110) = 3.9959 MD = 12.55 TD = 9.15 OP = 10.99 F = 5.23 E = 6.12	NS Orig CSE MD = 14.40 10.70 TD = 11.85 6.44 OP = 8.87 13.12 F = 6.11 4.35 E = 7.66 4.58

Table 4.10: ANOVA Results for NASA TLX Interface by Dimension Analyses

Workload Composite Means

The results of the ANOVA for the composite (weighted) workload means produced significant results only for the dimensions main effect (Table 4.10): F(1,110) = 3.99, MSE = 233.68, p = .0046. The mean weighted workload dimensions were, from highest to lowest, Mental Demand (12.55), Own Performance (10.99), Temporal Demand (9.15), Effort (6.12), and Frustration

(5.228). Although not statistically significant, the original interface (9.78) had a higher mean composite workload score than the CSE interface (7.85).

The hypothesized reduction in Mental Demand for the CSE interface did occur, but the results were not significant. The interface by dimension interaction mean composite scores indicate the shift in dimension ranking similar to those seen with the unweighted means. In this case, Mental Demand is rated highest for the original interface (14.40), but Own Performance is rated highest for the CSE interface (13.12). Figure 4.11 illustrates the nature of the shifts in the sources of workload due to the interface used. The primary difference in workload between the original interface and the CSE interface occurred in the Temporal Demand and Own Performance dimensions. The CSE interface appears to reduce workload associated with time pressure (TD), mental demand (MD), Effort (E), and Frustration (F) and shifts the source of workload to the decision-makers' own performance standards.

To further explore this shift, one-way ANOVAs were done for each of the dimensions. The results of all five ANOVAs were significant (Table 4.11). The largest effects ($p < .0001$) were associated with Temporal Demand, Mental Demand, and Effort in that order. The CSE interface resulted in lower weighted scores for all dimensions except Own Performance. This appears to support the limited capacity theories of human workload predicting that users will trade off one source of workload (or stress) as load is increased in another. For example, decision-makers will relax performance standards (Own Performance dimension) when time pressure (Temporal Demand) increases or the task becomes more complex (Mental Demand or Effort).

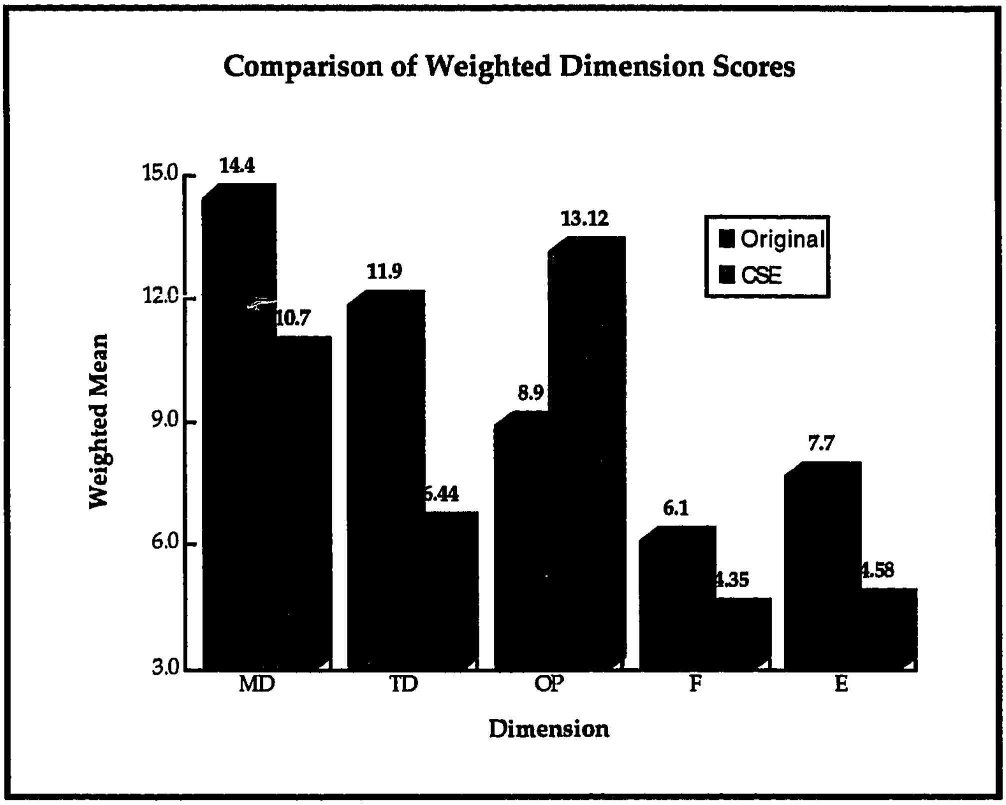


Figure 4.10: Comparison of Individual Weighted Workload Dimensions

Mental Demand	Temporal Demand	Own Performance	Frustration	Effort
p < .0001 MSE = 47.97 F(1,286) = 20.59	p < .0001 MSE = 30.80 F(1,286) = 68.59	p < .0014 MSE = 125.46 F(1,286) = 10.37	p < .0326 MSE = 48.22 F(1,286) = 4.61	p < .0001 MSE = 17.45 F(1,286) = 39.23
Original = 14.40 CSE = 10.70	Original = 11.85 CSE = 6.44	Original = 8.87 CSE = 13.12	Original = 6.11 CSE = 4.35	Original = 7.66 CSE = 4.58

Table 4.11: ANOVAs for Individual Workload Dimensions

Usefulness and Usability Measures

The remaining subjective measures were developed to assess the relative usefulness (or utility) of the interface windows in supporting the four tanker replanning decision tasks: problem identification, situation assessment, option evaluation and option selection. After each interface session, participants evaluated each window with respect to its usefulness in supporting each of the decision phases. The information was also used to develop profiles of the data view preferences of the decision makers. This section presents the results of the analyses of individual window usefulness and the decision-makers' display viewpoint preferences.

Window Usefulness for Decision Tasks

The hypotheses regarding specific windows in the interfaces primarily addressed preference shifts based on decision phase and interface. Since the ***Option View*** window was only available in the CSE interface a two-way ANOVA (interface x window) was not feasible. Instead, one-way ANOVAs were performed for both interfaces to discover the individual window preferences in each interface condition. Tukey-Kramer HSD (Highly Significant Difference) analyses were performed to determine the significantly different windows. This was followed by examination of specific windows through a series of one-way ANOVAs for selected windows by decision task. Finally, one of the graphic overview windows common to both interfaces, the ***Marquee*** was analyzed with a 2 (interface) by 4 (decision phase) ANOVA.

The first ANOVAs analyzed the summary scores (across all decision tasks) for each window. Because of the different number of windows in the two interfaces, these were conducted as two one-way ANOVAs of the summary scores by window. The results of the ANOVA for the original interface indicated significant differences between the windows: F(6,77) = 14.35, MSE = 372.20, p < .0001. Results for the CSE interface were also significant: F(7,88) = 15.15, MSE = 325.34, p < .0001. The means for each window are graphed in Figure 4.11, grouped by window to allow comparison by interface. Table 4.12 presents the results of the Tukey-Kramer HSD tests to establish which windows produced significantly different scores from the others. These initial results were used to focus the remaining analyses.

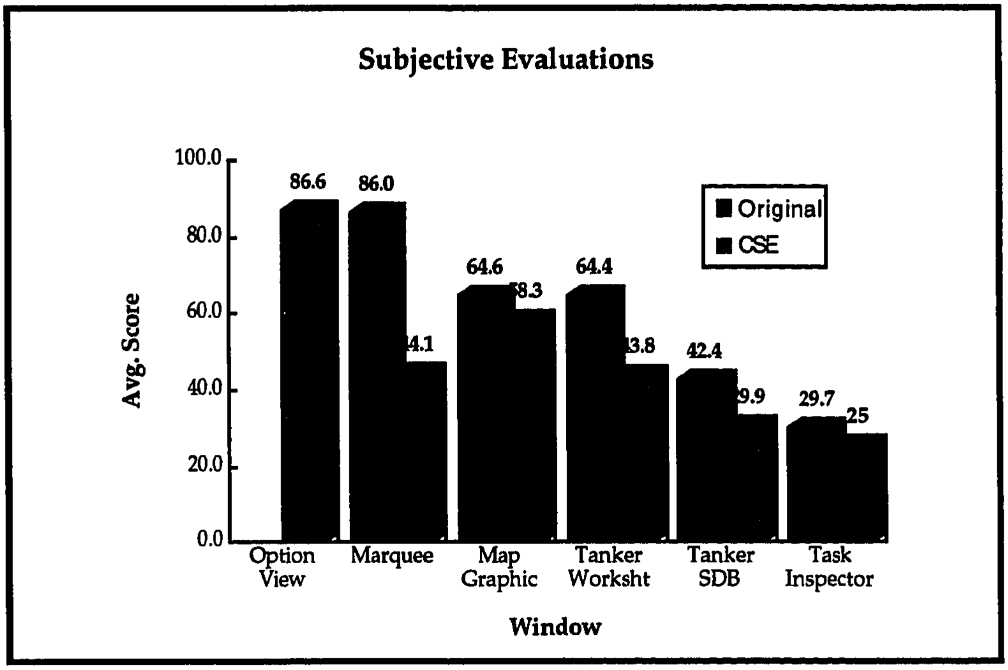


Figure 4.11: Subjective Evaluations: Summary Scores for Windows

Window	Significantly Different Windows	
	Original Interface	**CSE Interface**
Option View	N/A	All Windows
Marquee	Replanning Options Tanker SDB Task Inspector Task Notify	Option View
Map Graphic	Task Inspector Task Notify	Option View Tanker SDB Task Notify Task Inspector
Replanning Options	Marquee Task Inspector Task Notify	Option View
Tanker Worksheet	Task Inspector Task Notify	Option View
Tanker SDB	Marquee	Option View Map Graphic
Task Inspector	Marquee Map Graphic Tanker Worksheet Replanning Options	Option View Map Graphic
Task Notify	Marquee Map Graphic Tanker Worksheet Replanning Options	Option View Map Graphic

Table 4.12: Tukey-Kramer HSD Results for the Summary Subjective Scores

The results of the ANOVA and Tukey-Kramer HSD analysis for the CSE interface supported the hypothesis that the ***Option View*** window would dominate the preference scores for the CSE interface. As graphed in Figure 4.12, the mean summary score for ***Option View*** (86.6) exceeded the mean scores of the other windows in the interface.[6] The ***Map Graphic*** had the second highest score

---

[6] Raw mean scores were normalized on a scale of 0 to 100.

(58.3); no other windows in the CSE interface exceeded a 50% score. Examination of the individual window means for both interfaces disclosed two other interesting effects. First, the mean summary score for the ***Map Graphic*** changed very little across the two interfaces. Second, the mean summary score for the ***Marquee*** window (86.0) in the original interface condition almost matched that of the ***Option View*** window (86.6) in the CSE interface condition. In the CSE interface condition, the ***Marquee*** score dropped dramatically (44.1). This seemed to support the hypothesis that decision-makers would prefer the ***Option View*** when available and substitute the ***Marquee*** window when using the original interface.

To further explore this relationship, the mean summary scores and mean scores of the four decision tasks for the ***Option View*** window in the CSE interface condition were compared to the same scores for the ***Marquee*** window in the original interface. It was hypothesized that the results of these paired comparisons would find no significant differences. Figure 4.12 graphs the five paired scores for the ***Marquee*** and ***Option View*** windows. As predicted, the results of the ANOVA comparing the scores found no significant difference in the means for either the summary scores or the four individual decision phase scores.

To further verify the substitution effect, the scores for the ***Marquee*** window were compared for both interfaces. If the substitution hypothesis were correct, it was assumed that the ***Marquee*** scores would differ significantly between the two interfaces. Results of the one-way ANOVAs comparing the ***Marquee*** scores for the two interfaces revealed significant differences for the mean summary scores and the four decision task scores (Table 4.13). The mean scores for Option Selection yielded the largest difference (50 points): F(1,22) = 20.98, MSE = 0.0715, p = .0001, mean original = 92.42, mean CSE = 42.42. The smallest difference (30.3 points) was found in the mean score for Problem Identification:

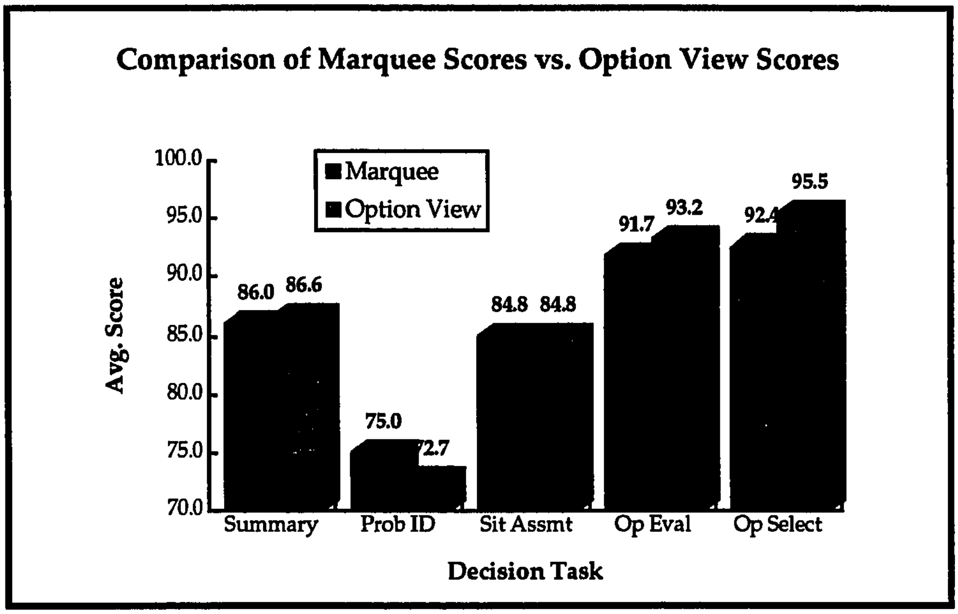


Figure 4.12: Subjective Evaluations - Paired Comparisons of Marquee (Original Interface) vs. Option View (CSE Interface) Scores

Decision Task	Original	CSE
Problem Identification F(1,10) = 8.47, MSE = 25.51 p = .0081	75.00	44.70
Situation Assessment F(1,10) = 14.12, MSE = 24.70 p = .0011	84.85	46.97
Option Evaluation F(1,10) = 20.65, MSE = 24.51 p = .0002	91.67	46.21
Option Selection F(1,10) = 20.98, MSE = 26.74 p = .0001	92.42	42.42
Summary Score F(1,10) = 18.31, MSE = 23.42 p = .0003	85.99	45.08

Table 4.13: ANOVA Results for Subjective Evaluations - Marquee Evaluations

F(1,22) = 8.45, MSE = 0.07, p = .0081, mean original = 75.0, mean CSE = 44.7. Consistent with the substitution hypothesis, these results suggest that decision-makers' preference in the support for the tasks changed depending upon the interface. Furthermore, the shift in preference to the ***Option View*** window in the CSE interface implies that this window more closely matched the task requirements of the decision-makers. Finally, the strength of this shift can be noted also in the change in the variability of the scores (Figure 4.13). In the original interface condition, the scores for the ***Marquee*** varied 17.42 points (from 75.0 to 92.42) across the decision tasks. In contrast, the ***Marquee*** scores in the CSE interface condition varied only 4.55 points (from 42.42 to 46.97).

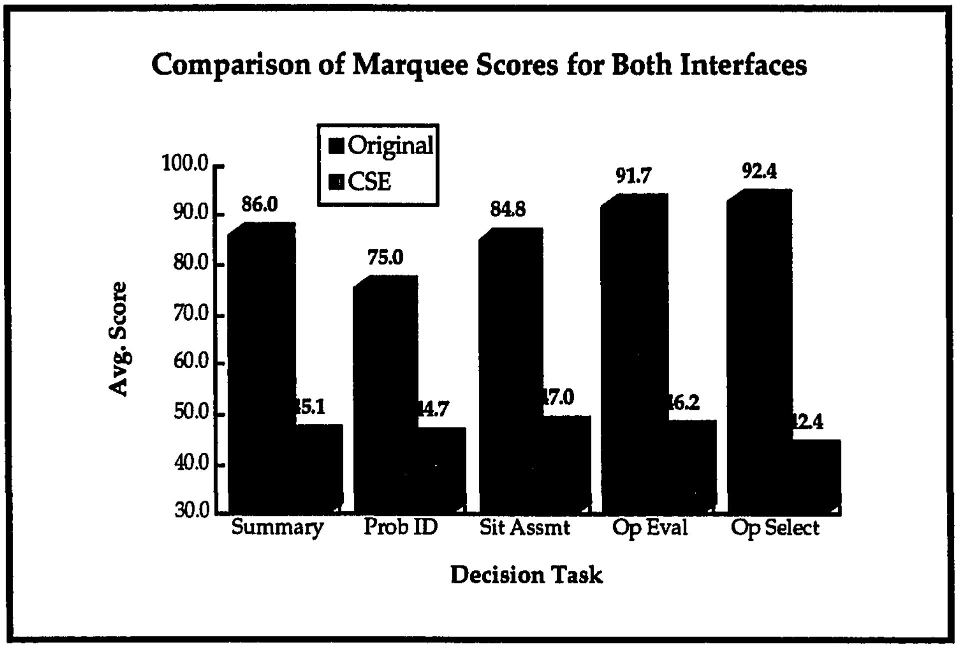


Figure 4.13: Subjective Evaluations - Marquee Scores Across Both Interfaces

User Preference for Display Viewpoint

The subjective evaluations also provided support for the hypotheses regarding the decision-makers' display viewpoint preferences. Previous analysis of the objective measures for the decision process examined the decision-makers' use of the windows during task performance based on selection and time spent viewing windows classified as graphic overview or detailed data displays. The analysis of those objective process measures supported the hypotheses that users would tend to select and use one viewpoint more than another depending upon the interface used. These findings raised an additional question regarding display viewpoint: were the decision-makers selecting windows based on their viewpoint preferences? This hypothesis would imply that the decision-maker's preferences changed depending upon the interface used. Since the decision trials were identical across the two interface sessions, it seemed unlikely that the change in window usage discovered was due to a change in preference. On the other hand, a consistent viewpoint preference across both interfaces might indicate that the usage difference was due to features of the interface design forcing the decision-maker to resort to less preferred views to accomplish the task. Such a counter-preference design could have implications for usability and system effectiveness.

Based on information gathered during the requirements identification phase, it was hypothesized that the decision-makers would prefer the graphic overview windows over the detailed data displays — regardless of the interface used. A 2 (interface) by 2 (viewpoint category) ANOVA was performed to assess the consistency of viewpoint preference across the two interfaces as reflected in the summed scores for each window and decision task. The category main effect was expected to be the only significant effect. The ANOVA results partially supported the hypotheses (Table 4.14). Results for the category main effect were significant, favoring the hypothesis: F(1,10) = 52.76, MSE = 5671.13, p < .0001, graphic mean = 30.05, detailed data mean = 17.25 (Figure 4.14a).

Measure	Interface Main Effect	Category Main Effect	Interface x Category Interaction
Viewpoint Preference	p = .0042 MSE = 912.60 F(1,10) = 8.409	p < .0001 MSE = 5671.13 F(1,10) = 52.76	NS
	Original = 25.27 CSE = 21.24	Graphic = 30.05 Detailed = 17.25	Orig CSE Graphic = 33.13 28.00 Detailed = 20.03 14.47

Table 4.14: ANOVA Results for Subjective Evaluations - Display Format Preference

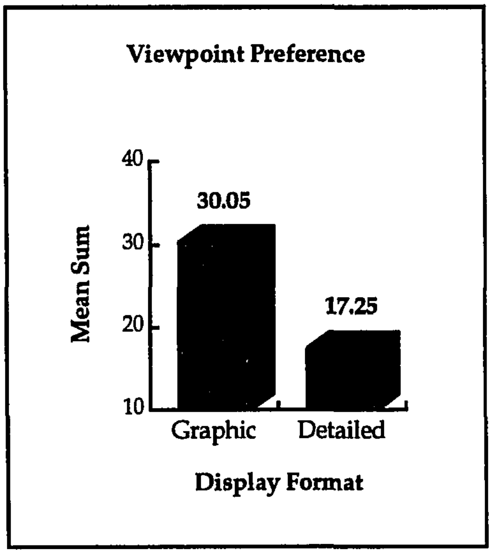


Figure 4.14a: Presentation Format Preference - Category Effect (p < .0001)

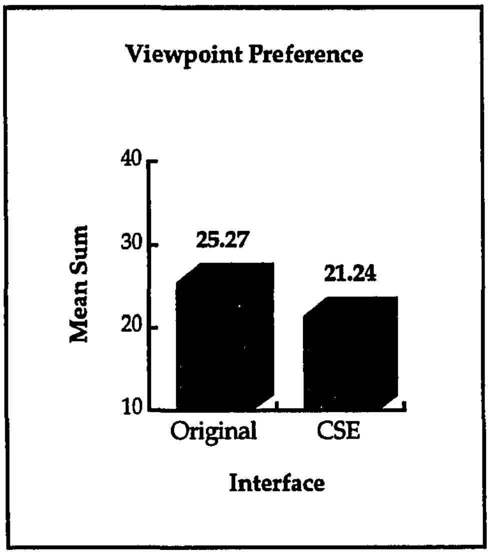


Figure 4.14b: Presentation Format Preference - Interface Effect (p = .0042)

An unexpected, statistically significant difference was found in the interface main effect (Figure 4.14b). The summed preference scores for the two interfaces were *higher* for the original interface than for the CSE interface: F(1,10) = 8.49, MSE = 912.60, p = .0042, mean original 25.27, mean CSE = 21.24. As noted in the discussion of the substitution of the ***Option View*** and ***Marquee*** windows, the scoring for the CSE interface tended to rate the ***Option View*** window at the top of the scale and assign substantially lower scores to all the other windows. The dramatically lower means on all the other windows reduced the overall interface score for the CSE interface. The resulting skew made the summed preference scores a poor indicator of overall preference for an interface. Nevertheless, general support for the hypothesis was borne out by the consistent preference for graphic versus detailed displays and an almost total absence of an interface by category interaction (p > .9).

The operational requirements to maintain adequate situational awareness and understand the interdependencies of the operational environment crossed all four decision tasks. For this reason, it was hypothesized that decision-makers would prefer graphic overview windows for all four decision tasks regardless of the interface condition. A 2 (interface) x 4 (decision phase) x 2 (viewpoint) ANOVA was performed to test this hypothesis. Based upon the hypothesis, only the viewpoint main effect was expected to exhibit a statistically significant difference.

Table 4.15 presents the ANOVA results for the main effects and interactions. The results of the analysis supported the hypothesis. The display viewpoint main effect produced the only significant result with the graphic overview displays (mean score = 6.44) preferred over the detailed data displays (mean score = 4.72): F(1,10) = 13.16, MSE = 169.47, p < .0001. None of the interaction analyses produced significant results; moreover, the interface by view interaction indicated an extremely close mean score match (p > .98). Figure 4.15 graphs the

Interface Main Effect	Decision Phase Main Effect	Viewpoint Main Effect	Interactions
NS	NS	p < .0001 MSE = 169.45 F(1,10) = 13.16	Interface x Phase x View NS Interface x Phase NS Phase x View NS
Original = 5.41 CSE = 5.33	Prob ID = 6.03 Sit Assmt = 6.13 Op Eval = 5.25 Op Select = 4.91	Graphic = 6.44 Detailed = 4.72	Interface x View NS F(1,10) = .001 MSE = .01 p = .9825 Orig CSE Graphic = 6.44 6.18 Detailed = 4.72 4.48

Table 4.15: ANOVA Results for Subjective Evaluations - Interface x Decision Phase x Display Viewpoint

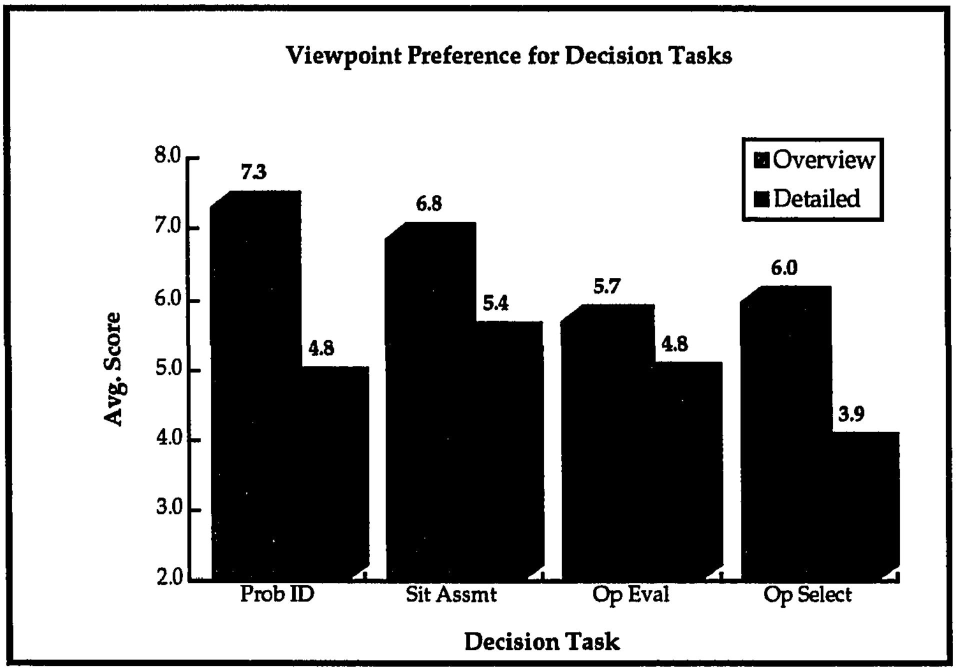


Figure 4.15: Subjective Evaluations - Viewpoint Preference for Decision Tasks

mean scores for graphic overview windows and detailed data windows across the four decision tasks.

General Usability & User Preference

The free-form interface evaluations and the final open discussions allowed the participants an opportunity to express opinions and make suggestions regarding the interfaces they used. Although formal analysis was not performed, the comments were consistent with the findings from the objective and other subjective data. Several general themes in usability and user preference are worth noting.

- Preference for the Graphic Overview Windows - Most participants commented on the usefulness of the graphic overview windows for "seeing what's happening all at once" and getting the "Big Picture to make the final choice." "The graphic display made the information highly digestible and greatly reduced the workload."
- Frustrations with the Original Interface - Subjects expressed frustration with the original interface for several reasons:
  - » necessary information was not available
  - » could not compare information across windows or options
  - » irrelevant information - "extra information -- information that is not needed;" having to "sort through the extraneous material"
- Frustration When ***Option View*** Removed - Subjects in the CSE-1st (CSE-Original) exposure order were frustrated by having to perform the second session trials without the ***Option View*** window. Frustration was expressed regarding
  - » *increased workload* - "Not having the ***Option View*** was more work;" "the answers were still fairly easy without any external factors, but the effort and mental demand increased."

  - » *required information is dispersed across several windows* - "made you work harder, because you had to go back and forth without the ***Option View*** screen;" "caused a lot of backtracking to find the info I wanted to make my choice."
- Ease of Use - Most participants mentioned that the two interfaces were easy to understand and use. In several cases, this comment accompanied admissions that the user "[knew] nothing" about computers.

As expected, the ***Option View*** window was mentioned more frequently than any other window. Comments focused on the following usability features:

- Ability to Comprehend at a Glance - "The ***Option View*** provided a clearer understanding of what was going on and what the choices involved. The information could be taken in quickly." "The most helpful window was the ***Option View*** because it graphically showed the relationships of the various air refuelings already scheduled to those that changed."
- Integrated Key Decision Factors - "The new screen (***Option View***) makes the interface much more efficient. It shows time and offloads remaining after the option, making decision-making much easier." "It had all required information on one screen -- no need to flip from one window to another."
- Eliminated Mental Calculations - Subjects indicated the ***Option View*** window was "useful in limiting amount of displays shuffled through and mental calculations." "The math was done already and it was a question of picking one of 2 effective options."

The strength of the preference for the ***Option View*** window was further supported by several users expressing a desire to be able to access mission detail directly via ***Option View***. Users also wanted to be able to perform "what-if" option comparisons using "drag and drop" direct manipulation to move receiver missions to different times and/or tankers. As indicated in the previous chapter on the FLEX Case Study, both features were part of the CSE-based design recommendations, but were not implemented for the initial prototype to preserve experimental control.

In group discussions conducted after the second interaction sessions, participants volunteered some information about their decision processes. For the most part, these comments echoed the information in the subjective evaluations. The subjects preferred using the graphic windows to achieve an overview of the problem and the related factors. When using the original interface, they were frustrated by the requirement to perform fuel calculations manually and having to hunt for the data they needed in several different windows. The participants unanimously endorsed the Option View window as the most useful view for the decision-making tasks.

The most notable comments came from the two groups that used the CSE interface first. Several participants in the CSE-first exposure order described using their "mental image" of the operational relationships in the ***Option View*** window when the window was no longer available (original interface condition). These users indicated that they transferred the remembered relationships to the almost analogous ***Marquee*** window and used that window to guide their search for the missing details.[7] Their description of the ways they used the original interface after using the CSE interface were consistent with the usage patterns discovered in the objective process measure analyses and the subjective evaluations of window usefulness. The use of a "mental image" described appears to support the previous interpretation of the interface by exposure order interactions.

---

[7] The ***Marquee*** window graphically showed the current flow of receivers against tankers across time, but required decision-makers to calculate the fuel remaining and mentally construct and compare the two options.

4.1.7. Discussion

The experimental portion of the CSE evaluation had two purposes. First, the experiment supported the system design evaluation phase required as part of the CSE framework. Second, the experimental comparison of the two interfaces served to evaluate the output, or product, of CSE-based development as part of an overall evaluation of the design method. Each of these functions is discussed below.

Evaluation as Part of CSE-Based Development

The CSE framework for systems design and development can be fully integrated into the traditional system development life cycle (SDLC). This integration includes multi-phased evaluation of the life-cycle products for feedback and control of the development effort. The experimental evaluation of the CSE interface design provided confirmation for several of the design improvement hypotheses developed using the CSE methods. Furthermore, input from the operational users supported several proposed, but currently unimplemented, interaction features (i.e., direct manipulation detail access, "drag and drop" option exploration, etc.). In the normal course of development, these findings would provide guidance for the next iteration of the design.

Testing in the early phases of design not only confirms and refines the evolving definition of the system requirements, but also helps identify the appropriate measures of performance (MOPs) and effectiveness (MOEs) by which the fully operational prototype or system should be evaluated. The experimental evaluation of the two FLEX interfaces set new benchmarks for task performance (speed and accuracy), decision processes (window changing in information search), and cognitive workload. Moreover, the early data analysis models can serve as templates to speed later evaluation. Parallel evaluation can significantly streamline and speed the overall development process by rapidly supplying results of interim evaluation to support iterative design.

The FLEX interface experiments demonstrated the feasibility and utility of conducting early evaluations with operational users. The experiments were conducted over a four-day period, involving only seven hours participation by the twelve subjects. All of the objective data were collected by the testbed software, eliminating the need for additional observers or data collectors. Furthermore, the entire testbed was highly portable and testing could have been accomplished easily at the operational site. Finally, although considerable analysis was performed post hoc, the key information on performance, process and workload was available in rough form within 48 hours.

Demonstrated Benefits of the CSE-Based Prototype

The results of the experimental evaluation favored the CSE HCI design for all the objective and subjective measures. Decision-makers using the CSE interface completed tasks faster with greater accuracy and used the interface to review information more effectively than when using the original interface. The difference in the source of workload from time stress to the more positive pressures of self-imposed performance standards also favored the CSE interface. Integrating the key decision information in the Option View window allowed the decision-makers access to the required information in the preferred graphical overview display format rather than wading through the detailed data. Finally, the users uniformly preferred the CSE interface for all the decision tasks involved in tanker replanning.

4.2. Evaluating the CSE Design Process

At the simplest level, evaluating a design and development process improvement is a comparison of the tradeoffs in potential benefits gained versus potential increases in development costs. The FLEX Tanker Operations Case Study was undertaken primarily to demonstrate the potential improvementsin end-user performance. Despite this product orientation, it is possible to examine certain aspects of the acquisition process employed. This section presents a brief discussion of two example metrics for assessing the cost/benefit tradeoffs involved in the CSE framework (Figure 4.16). The first section examines the CSE development process in terms of the potential changes in resource requirements. The second section presents a study to assess the CSE development process products (i.e., requirements documents, design prototypes, etc.) by examining the change requests submitted on the three prototypes developed at Rome Laboratory as part of the initial technology demonstration effort.

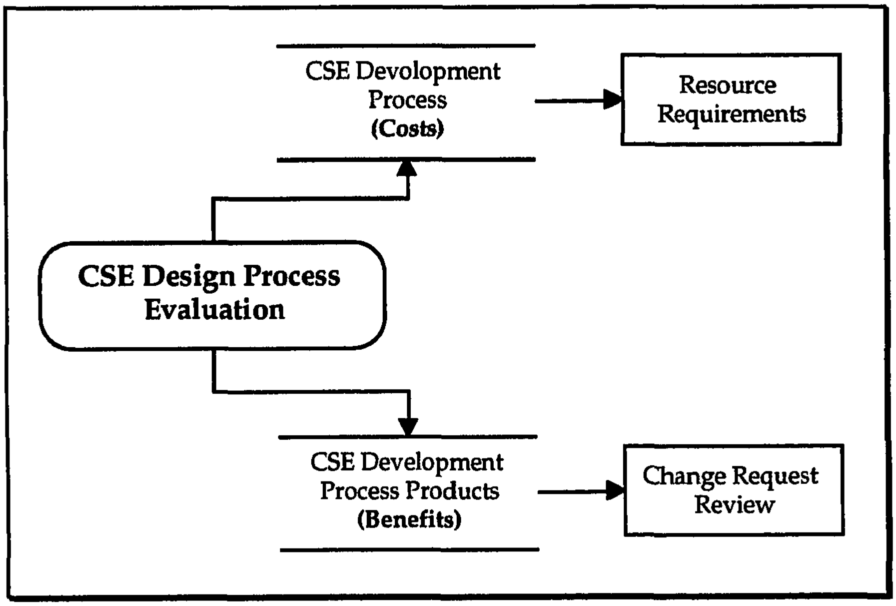


Figure 4.16: Organization of the CSE Design Process Evaluation

4.2.1. Evaluating the CSE Development Process

Planning and control of system development efforts tracks projects against schedule and budget constraints. Thus, the potential performance and functionality improvements achieved with the CSE framework must be balanced against the potential cost and schedule impacts of implementing the framework within the development process. Organizations considering incorporating the CSE framework in their development process need answers to three fundamental questions:

- Do we have to hire a CSE specialist?
- Can we use the CSE requirements identification methods with the information usually available to us?
- Will CSE take more development time?

The FLEX Tanker Operations Case Study development was conducted independently of the Rome Laboratory (RL) FLEX development effort, making comparisons of the two processes difficult. The creation of the functional prototypes combined the efforts of four RL designers and the contracted services of two large organizations. Developing the three versions of the prototype was the primary activity of the RL designers during an 18-month period. In contrast, the case study represents the efforts of a single individual working part-time on a small subset of the overall FLEX system. Despite these significant differences, the author's case study effort is comparable to the contribution of a development team member and furnishes several useful observations.

Personnel Requirements

Do we have to hire a CSE specialist?

One obvious cost component involves the personnel required to accomplish the CSE analysis and design activities. CSE-based development may be accomplished by the *typical system development team*. Ideally, CSE activities should be integrated into the overall development process and, thus, become the concern of the entire design team. In practical application, this integration will involve certain team members more than others. In some organizations, many of the activities might be the responsibility of someone with a human factors or similar background. However, it should be noted that the key CSE activities -- requirements modeling and system design -- are more naturally assigned to designer/developers. The CSE Design Practitioner's Handbook (Appendix A) provides practitioners a readily digestible, step-by-step means for incorporating findings and observations from the empirical literature into their understanding of the system requirements and design solutions. With training and practice, the key CSE activities at each development phase should be within the capabilities of moderately experienced designers with *an active interest in the end-user and operational context*.

Information Requirements

Can we use the CSE requirements identification methods with the information usually available to us?

As demonstrated in the discussion of the FLEX Case Study effort (Section 3), the CSE requirements identification activities involve the same general information resources typically used by the development team:

- *Document Review* - white papers, mission needs statements, operational manuals, system documentation for interconnecting systems, etc.;
- *Interviews* - sponsor/client, senior operational personnel, end-users; and
- *Observation* - field observation of operational environment, procedures and systems currently in use, etc.

The FLEX Case Study used the same information resources employed by the FLEX development team. In fact, despite extraordinary cooperation from the RL development team, the contractors, and the FLEX Working Group (FWG), the author had significantly less access to the most pertinent information. For example, much of the early design work at RL was based on extensive on-site interviews and observations at operational bases in the US, Europe, the Pacific, and the Far East. The author was provided copies of the trip report summaries. The development team had extensive access to the operational representatives on the FWG. The author had a few brief conversations with some of the officers involved, but was not able to sit in on contractual meetings between RL, the FWG and the contractors. These meetings constituted the primary means for communicating and addressing requirements details and change requests. Nevertheless, the author was able to use this subset of information to complete the CSE requirements and design activities for the case study. It seems clear that a fully integrated team member would have access to sufficient information to accomplish the same goals.

Time Requirements

Will CSE take more development time?

Since CSE activities flow within the normal activities of the system development process, there should be no negative impacts on the schedule. More importantly, CSE-based development should help to *reduce* the overall development time. As discussed in the next sections, the primary means for this time savings is in the reduction of rework through early identification of a more accurate set of operational and end-user requirements combined with support for developing design solutions based on those requirements.

As indicated previously, the requirements, design and evaluation activities for the FLEX case study were accomplished by the author working part-time over a period of approximately twelve weeks for a total of 198 labor-hours (LH). As indicated in Table 4.16, the time involved in requirements identification and design totaled approximately 85 LH; however, during that period, an equivalent (or greater) amount of time was spent researching the generic CSE guidelines information that is now contained in the CSE Design Practitioner's Handbook. The programming of the interactive prototype, including the CSE prototype and the re-creation of the six screens from original FLEX interface, was accomplished in approximately 48 LH. This phase also involved the programming of the automated data collection functions. Finally, the design and implementation of the experimental evaluation of the interface prototype was accomplished in three weeks using 65 LH. This included all the administrative activities involved in arranging in-house research using tanker squadron officers from Griffiss AFB. Although the final data analysis represented in this chapter required several weeks, the key feedback on the prototype was available almost immediately.

Development Activity	Weeks	Labor-hours (LH)
Requirements Identification & Design (Total effort less time spent on generic CSE Design Practitioner's Handbook research)	7	85
Prototype Development	2	48
Experimental Design & Data Collection	3	65
TOTAL	12	198

Table 4.16: Time Required for FLEX Case Study Development Activities

In summary, integrating CSE methods into the development process should not require substantial changes in personnel, information resources, or development time. CSE development activities are designed to complement and support good systems engineering practices. Furthermore, the benefits received from integrating CSE methods are contingent upon the quality of the overall development process. CSE methods cannot "rescue" projects lacking sound planning backed by adequate feedback and controls.

4.2.2. Evaluating the CSE Development Process Products

This section presents the study performed to evaluate the potential contribution of the CSE framework towards identifying a more robust set of HCI requirements. The section is divided into three parts to present the study design, procedures, and results.

Design and Hypotheses

In most cases, the human decision-maker's cognitive task requirements (CTRs) are implied, but not stated, in the system requirements documents. These implicit requirements may not be incorporated into design and are unavailable for review and inspection against the design. Failure to fully identify and represent the human cognitive requirements impacts development, in part, as change requests (CRs) requiring redesign and rework. To evaluate the benefit of the CSE framework for requirements identification, the revised requirements in Appendix F were compared to the changes requested in the three prototype iterations of the Rome Laboratory 6.3a development effort.[8] The CSE framework was predicted to result in changes in identified requirements which, if incorporated into the system design, would have removed the necessity of many of the later CRs. Viewed from the designer's perspective, the CRs reflect "misses" in the requirements identification process. Thus, the potential reduction in rework is predicted in the extent that the revised (i.e., CSE requirements) "hit" those previously missed requirements.

---

[8] The case study's focus on the FLEX Tanker Module did not involve a complete revision of all the FLEX windows based on the CSE revisions to requirements. For this reason, it was not possible to have the FLEX Working Group (FWG) review the CSE Interface and generate a matching set of change requests for that revised interface.

Study Procedures

In the period from September 1992 to October 1993, FWG reviews of FLEX Prototypes 1, 2, and 3 surfaced 174 specific change requests. These were compiled and addressed by the Rome Laboratory development team. Many resulted in direct changes to the system design for the next iteration of the prototype. In December 1993, the author received a copy of the Change Requests logged at Rome Laboratory up to October 1993 (Appendix I).[9] These CRs were compared to the revised System/Segment Specifications (SSS) requirements document developed using the CSE framework (Appendix F).

For this study, the author first coded each of the CRs as belonging to one of three categories:

1. **Functional Task Requirement (FTR)** - system requirements, including CTRs, associated with the decision-making tasks (i.e., information search, structuring, communication, etc.);[10]
2. **Interaction Task Requirement (ITR)** - system requirements associated with the user's operation of the interface; or
3. **Other (OTH)** - system requirements external to the user's tasks (i.e., import/export capabilities, etc.).

The coded CRs were then matched to the CSE requirements document and scored as present (1) or absent (0). ITRs representing universally prescribed HCI standards were judged to be incorporated in the CSE framework. Additional discussion of these ITRs can be found in Section 4.2.3 below. More specific ITRs were compared to the CSE requirements document and scored accordingly. The OTH CRs fall outside the specific focus of this study; nevertheless, these

---

[9] The CSE prototype was developed in July 1993 and tested in September and October 1993.

[10] This category principally comprised the CTRs, but also included functional requirements associated with the decision-making organization (e.g., coordination support) that extended beyond the tasks of the individual duty officer.

requirements were also scored against the CSE requirements document and the CSE framework.

Results

The requirements documents present information at a level of abstraction above that of the individual change requests. As a result, scoring the CRs against the CSE requirements document involved a subjective judgment regarding the underlying requirement implied by the CR. This judgment required not only knowledge of the domain, but also considerable knowledge regarding the system features referenced. In addition, the relationship of requirements to design and subsequent change requests does not involve a one-to-one correspondence. These aspects are discussed further below.

Table 4.17 presents the results for the change request review. **CR Count** presents the number of change requests in each category; **Match Count** presents the total in each category judged to be represented in the CSE requirements document. **Percent Coverage** is a ratio of **Match Count** to **CR Count**. Similarly, **Total Coverage** is the ratio of **Total Match Count** to **Total CR Count**.

Change Request Category	CR Count	Match Count	Percent Coverage
FTR	122	97	80%
ITR	43	23	54%
OTH	9	0	0%
Total	174	120	69%

Table 4.17: Change Request Review Summary

Table 4.17 indicates that the most significant coverage of CRs was associated with the support of functional (i.e., decision-making) tasks. Here, the CSE framework resulted in an 80% (0.795) "hit" rate. In several cases, the CRs specifically restated needs expressed previously in both written and verbal form by the user community (that is, the FWG and the operational representatives consulted in the requirements interviews). Furthermore, there were some issues that continued to surface in reviews of subsequent prototype versions. For example, a request for the capability to "compress" the information (more missions with less detail) is a specific instance of the more general CTR to provide the means for allowing the user to "step back" to the appropriate level of abstraction through user-definable views. While the request for user-tailorable *query* results was addressed in the subsequent prototype, the underlying CTR was not and resurfaced in several requests for some means of extending the view while reducing the detail. Each instance required an additional change.

The 54% (0.535) rate of ITR coverage reflects the level of detail associated with the CRs in that category. These CRs involved low-level interface features (e.g., naming and placement of buttons) not appropriate for representation in requirements documents. As indicated in the discussion of scoring categories, CRs related to commonly accepted HCI standards (e.g., consistency of interaction across the interface) were deemed as represented in the CSE framework to the extent that these CRs would not be associated with a CSE-based design. Thus, when the CR referenced a need to know when the system is "working" (as opposed to locked-up), with a message or animated cursor, that CR was scored as represented in the CSE framework. However, a request to change the order of two columns in a specific window could only be captured during direct review of the window in question as there are no commonly accepted design principals that would dictate this ordering. Furthermore, while the CSE framework advocates the identification of these layout issues during the earliest phases of design before the changes become more costly, such detail is not specifically referenced in the revised requirements documents or associated models.

The final category, OTH, represented requirements deemed outside the scope of the case study. These included details of system import/export capabilities that should be captured in the system requirements. Due to the focus of the case study, none of these CRs were addressed in the CSE requirements revisions.

Of the 174 CRs identified by the FWG, the CSE framework claimed identification of 120 for an overall coverage of 69% (0.6896). It should be noted that this study did not provide the means for assessing the extent that a CSE-revised design would have 1) differed in addressing the CRs raised by the original design or 2) reduced the number of new CRs. Despite these caveats, the results suggest some interesting points. The preponderance of CRs (70%) categorized as functional task requirements (FTRs) mirrors the priority of these requirements with respect to the system's ability to support the mission. Furthermore, changes in these requirements typically entail the most significant (and most expensive) changes to the system, potentially including revision of the entire system concept. It is, therefore, notable that the CSE framework appears to provide substantial support (by virtue of the 80% coverage) to the identification and clarification of these critical requirements. In addition, these results indicate the integral role of the information presentation and interaction design in the support of these functional tasks and suggests that such design issues should not be considered equivalent to the more superficial aspects of interface design.

Change requests are an inevitable part of the prototyping and development process, particularly in the design of large, complex systems. The goal of the CSE framework is not to eliminate changes, but to reduce the *impacts* of changes on the system development process. Review of the CR coverage suggests that the CSE framework enhances the developer's ability to capture many of the critical system requirements driving the design process. Furthermore, since the framework focuses on core system concepts in terms of support to mission accomplishment, CSE-based requirements should generate designs that require less rework in these fundamental areas.

4.3. Summary

This section reviews the results of the multidimensional evaluation performed to assess the value-added of the CSE framework for HCI requirements identification, design and evaluation. The section concludes with suggestions for further validation of the design process and products.

4.3.1 Evaluation of the CSE Framework

The evaluation of the CSE framework examined the *benefits* of the using the approach in terms of the quality of development process products versus the *costs* in development time and resources. Evaluation was accomplished using a multidimensional framework of objective and subjective measures analyzing data collected from documentary sources and a controlled experiment. The results of each investigation are presented below.

The CSE Design Product Evaluation

Evaluation of CSE-based HCI design comprised investigations of decision-making performance and processes coupled with process and performance assessments for the decision support aspects of the HCI design. The analysis was based on objective and subjective measures collected in a controlled experiment involving domain-knowledgeable users. As discussed in the previous section, the CSE-based HCI design consistently produced more desirable decision-making performance. Subjects using the CSE interface arrived at decisions approximately 26% faster with a 12% improvement in decision accuracy. Analysis of the interaction between the interface used and the exposure order revealed significant changes in performance which may be due to support in the

CSE interface for creating a returnable mental model of the complex interdependencies in the operational environment. The CSE interface users demonstrated more focused use of the interface for information review as reflected in 29% fewer window changes. Further review of the window usage revealed several trends regarding the use of graphical overview displays versus detailed data displays. The interaction effects noted in the objective performance measures were also significant for the objective process measures.

The decision support provided by the CSE interface resulted in a 20% reduction of task workload as measured by the NASA Task Load Index. The TLX analysis further revealed a shift of the source of workload from external stressors (mental and temporal demand) to the internal motivation factors measured as the user's own performance standards. Subjective evaluations of the individual interface windows provided additional support for the objective process findings regarding graphic overview and detailed data displays. Users uniformly rated the graphic overview windows higher than detailed data displays across the four decision tasks (problem identification, situation assessment, option evaluation, and option selection). When available, the ***Option View*** window received the highest scores; when ***Option View*** was unavailable, users rated the ***Marquee*** highest. This scoring shift matched the window selection shift noted in the objective process measures under similar circumstances.

Overall, the evaluation demonstrated the benefits of using the CSE framework in four key areas:

- *System Development Process*
    - » Uses currently available development resources -- no additional development costs

  - » Supports the identification of a more robust set of requirements to reduce the potential costs of re-work
- *Decision-Making Performance*
  - » Reduces decision performance time
  - » Improves decision accuracy
  - » Supports more focused, effective use of the interface
- *Decision Support Performance*
  - » Reduces workload overall
  - » Shifts source of workload to more positive internal performance standards
- *User Acceptance*
  - » Focus on cognitive task requirements results in a better match with the decision-maker's information presentation preferences.

The CSE Design Process Evaluation

Two metrics were developed to assess the cost/benefit tradeoffs involved in using the CSE design process. The first examined the CSE development process in terms of the potential changes in resource requirements. The second involved a study to evaluate the CSE development process products (i.e., requirements documents, design prototypes, etc.) by examining the change requests submitted on the original prototype interfaces.

Investigation of the CSE resource requirements considered three basic cost sources: personnel, information resources, development time. The results of each are summarized below.

- Personnel Requirements - The framework presented in the CSE Design Practitioner's Handbook is designed to support moderately experienced designer/developers with an active interest in the human-computer interaction design. No "CSE specialist" is required.

- Information Resource Requirements - The CSE requirements identification activities involve the same general information resources typically used by the development team. No "special CSE information" is required.
- Development Time - CSE-based development is designed to *reduce* the overall development time. No additional development time is required.

Review of the original FLEX interface change requests against the CSE-based requirements document suggested the CSE framework may enhance the developer's ability to capture many of the critical system requirements driving the design process. The CSE framework produced revisions to the original FLEX requirements document that captured 80% of the functional task requirements (including the cognitive task requirements) and 54% of the interaction task requirements for a mean coverage of 69% overall.

4.3.2. Areas for Further Research

Evaluation research begins with initial hypotheses about which aspects of the process or product of interest will provide the most leverage for improvement. Investigation of these early hypotheses clarifies the questions and helps to identify the most appropriate means of obtaining reliable answers. In addition, analysis of the research results may prompt reassessment of the priority placed on certain features. Finally, each phase of evaluation usually surfaces new issues for investigation.

Design Product Research

The evaluation of the CSE interface prototype demonstrated the value of certain interaction design features for improving decision task performance.

Allowing decision-makers to visually compare options reduced the task workload associated with mentally simulating outcomes of a given option and the ripple effects across the operational network. The ***Option View*** window permitted decision-makers to use their preferred mode of information review to maintain a desired level of situational awareness. The decision-makers were consistently more accurate in choosing the "better" of two options; however, the options were artificially limited. Furthermore, support for decision-maker option generation and exploration was not addressed in this study.

The interaction effects noted in performance using the original interface after prior exposure to the CSE interface seemed to suggest the presence of a retained model separate from general learning effects. This was borne out by the fairly level performance achieved using the CSE interface in either exposure order. Data were not collected to assess the content of task and interface learning from session to session. None of the measures developed attempted to ascertain the content of the decision-makers task and domain models before or after interaction with interfaces. Further research is required to more accurately partition the task and interface learning effects from the closely related mental model of the operational environment.

Finally, although there was some evidence in the subjective evaluations and open discussions that the decision-makers were using the key criteria identified by the experimenter to choose their preferred option, no data were collected on why the decision-maker preferred one option over another in each trial. This information is critical in determining whether the HCI design did, in fact, direct the decision-maker's focus as intended. Furthermore, it helps to validate the general applicability of the chosen criteria. Additional investigation of this factor could be integrated with a study on the content of the decision-maker's mental models of the task and operational environment.

Design Process Research

The evaluation of the CSE design process identified several aspects of the potential cost/benefit balance that require further investigation:

- Can the average motivated designer use the CSE method to consistently develop a more accurate set of system requirements?
- Will those improved requirements routinely lead to designs that deliver more functionality and better performance to the operational end-users?
- Can the process and output of the CSE-based development be appraised using accepted software engineering metrics?
- Can the assumptions regarding resource requirements be demonstrated with traceable dollar savings in personnel costs for initial development and revision?

Further validation of cost savings would involve conducting case studies using similar resources (personnel, information resources, and technical support) to accomplish similar tasks. As with system testing, the validity of any findings would hinge on the representativeness of the subjects and development tasks. Software engineering metrics for HCI design quality are virtually non-existent. Developed code may be examined for efficiency, but determining the effectiveness of the delivered system still relies on an assessment of the cooperation between the human user, the computer, and the operational tasks. To apply metrics to the quality of the development process without a means for assessing the value of the output would present an unbalanced view of the true cost/benefit. The next phase of the CSE methodology needs to better integrate task performance metrics with software performance metrics.

5. Summary and Research Agenda

This research began with the premise that the human decision-maker's cognitive tasks constitute an important dimension of human-computer interaction (HCI) requirements in systems built to support human-computer cooperative decision-making. While some of these cognitive task requirements (CTRs) are recognized by the HCI design community, they have been poorly represented in requirements specifications. Failure to incorporate CTRs in software requirements breaks the necessary links between requirements and design that ensure traceability and preserve the original concept of the operational need through the later phases of development. Furthermore, these cognitive requirements will not be included in the iterative evaluation of designs.

Cognitive systems engineering (CSE) presents a multidisciplinary approach to HCI design for human-computer cooperative decision-making. More than vague concepts of "user-friendliness," identifying HCI design features that support the cognitive aspects of the decision-maker's real-world tasks provides system designers new leverage for improving decision performance and increasing user/sponsor satisfaction. Although a number of studies have been done in recent years to illustrate the value of incorporating CSE principles in design, the methods used to produce CSE-based HCI designs were outside the expertise of the average design team. Designers need a straightforward, cost-effective procedure for identifying CTRs, representing them in requirements specifications, translating them into HCI design concepts, and evaluating the efficacy of the implemented HCI design. The research set forth here proposed a framework for incorporating information from the standard requirements sources into models and specifications of cognitive task requirements. The FLEX Tanker Module Case Study demonstrated the application of the CSE Design Framework to a complex, real-world system development effort. The subsequent evaluation of both the development process and the developed product (the HCI design) supported the hypothesized improvements to requirements specification and decision performance.

This research also helped to identify several unresolved issues in the CSE Design Framework that must be addressed to realize its potential in the current system development environment. The remainder of this chapter summarizes the status of the CSE Design Framework and outlines research and development areas that may prove fruitful.

5.1 Integrating HCI Design into the System Development Effort

One of the persistent problems plaguing HCI design has been the continued relegation of human-computer interaction issues in system design to considerations of interface operation (menus and navigational aids) and screen layouts. The presumption in this design model is that HCI represents only those surface aspects of the system, rather than being intimately connected to the user's ability to exercise the functional processes of the system to accomplish real-world tasks. One goal of this research was to provide further support to endorse the ***functional*** role of HCI in assuring system performance. To secure a stronger role for HCI early in system development will require flexible, practical and economical methods for integrating HCI issues with the software and hardware design activities at each phase of system development.

5.1.1 Integration Goals

Effective use of the CSE Design Framework requires the ability to create and integrate the informal conceptual models used to map out the CTRs. As presented in this research, the links between models and links from the models to other system development activities are extremely weak. Part of the difficulty lies in nature of informal models. Although more formal structures standardize the integration between models and support automated analysis, the representational restrictions are inappropriate for early conceptual models of the user, tasks and context. Moreover, standardizing the modeling approach eliminates some of the ability to capture essential domain realities and may make the entire CSE Design Framework too inflexible for designers to use effectively.

Informal models have long been part of the early phases of software and HCI design. Despite the considerable power available in computer-aided software engineering (CASE) tools, they do not support the construction and maintenance of the early models that underlie the designers' concept of the system in a form that truly integrates them into the formal models or requirements specifications. For this reason, the software and HCI design literature continues to explore techniques for developing and employing conceptual models in design (c.f., Ehrhart & Aiken, 1991; Montazemi & Conrath, 1986; Zahniser, 1993; and Zhang *et al*, 1992). Integrating informal models implies creating a framework of linkages between related models such that information may be shared among the various designers and across design functions (including the hardware and software designs). Until conceptual models are integrated and preserved in design documentation it will not be possible to trace design features back to the originating requirements.

5.1.2 Linking HCI Design and Evaluation to the SDLC

Chapter 2 presented the CSE Design Framework for HCI prototype design and development modeled within phases of a standard system development life cycle (SDLC). One of the cornerstones in the CSE Design Framework is the feasibility of integrating HCI design concerns into hardware/software development. Critical links between HCI design activities and the overall development effort are located in each phase of development. Cementing those links requires the ability to share early problem definition models across design teams and to incorporate information from those models into the requirements models developed for hardware, software, and HCI designs. This multi-directional flow implies incorporating a better understanding of hardware and software requirements into the HCI design concepts and vice versa. In addition, it is necessary to establish verifiable, traceable methods for stating cognitive requirements in system requirements specification.

Moving from requirements to design concepts presupposes methods for matching attributes of the requirements to configurations of HCI design features. Efforts to support this matching are currently underway as part of the research funding the FLEX Case Study. The Knowledge-Based Workbench for HCI Design employs case-based reasoning and COTS software to assist in requirements development and HCI design exploration in a series of linked templates. The identified requirements narrow the design options and provide suggestions for possible configurations. In addition, the requirements form the basis for constructing evaluations of prototype designs.

Although achieving a better match to delivered functionality is clearly important to system designers, the predominant feasibility criteria is the cost/benefit tradeoffs involved in enhancing the representation HCI design factors in the standard development processes. The design process evaluation presented in this research suggests that a positive ratio may be achieved between benefits and costs without major adjustments to the system development process. The thumbnail evaluation did not address many of the essential cost/benefit factors that would influence general acceptance of the method. For example, no economic measures were presented to compare the costs of developing either the original FLEX interface or the CSE-based enhancement. There was no rigorous validation or verification of either the original or CSE-enhanced requirements. Moreover, the case study provided no mechanism for investigating whether either set of requirements reliably represented the predictable outcome of the requirements identification process that produced them. This issues and others remain open for further investigation.

5.2 Research Agenda for Developing and Validating the CSE Design Framework

5.2.1 Improving Model Integration

It may be most useful initially to explore methods for converting conceptual diagrams into more standard analysis forms, such as those described for problem analysis in Davis (1993). Davis describes a set of problem analysis primitives for partitioning, abstracting and projecting the objects, functions, and states that comprise problem definitions. Although this work does not describe methods for transitioning from these primitives to formal representations, the categories (objects, functions, and states) each suggest certain modeling techniques. It may be possible to use these relationships to create pathways for integrating conceptual models into more formal representations. True integration will allow traceability from design features to requirements specification and back through the formal models to elements in the original conceptual models. For this reason, validation of an integrated modeling framework should incorporate a physical trace, as well as cross-indexing and other hierarchical analysis methods.

The most logical first step in finding an implementation path would appear to be adapting a concept-organizing COTS tool to export models in a standardized form for interpretation by a CASE tool. Concept organizers usually provide a text export feature; however, the output of that process is not user-customizable. To manipulate it requires modification of the initial representation; thus, using this approach would impose unnatural constraints on the modeling process. Another approach would be to introduce a filtering process between the text output and the CASE tool. Both of these approaches may be explored through small studies that will help to identify the relevant dimensions of prototypical informal models, determine the content and form of links between models, and match these to the input requirements of the various formal models supported in CASE environments.

5.2.2 Establishing Links Between Design Goals and Guideline Literature

The CSE Design Framework was conceived in part as a means for establishing a better linkage between requirements, design goals and the HCI design guideline literature. Several researchers present credible links between the basic cognitive research and HCI guidelines (c.f., Gardiner & Christie, 1987; Smith & Mosier, 1986); however, these guidelines generally consider only the system operation aspects of HCI design. Furthermore, their organization and presentation treats cognitive requirements as discrete elements rather than interdependent dimensions of larger requirements models. A more powerful construct would be to link guideline literature to design goals and, thus, to configurations of requirements. This is the ultimate goal of the CSE Design Framework.

Several avenues of research appear promising for further extension of the CSE Design Framework. The modeling research described above will help to identify an initial set of prototypical requirements configurations. For example, these primitives might include conceptual models of the fundamental aspects of situational awareness, including variable parameters (e.g., decision horizon, threat, etc.) common to all situational awareness requirements. Using the parameters from the tables in Appendix B, it may be possible to create links from these configurations to the basic cognitive research and from there to the guideline literature. Another approach may be to construct the links directly to both HCI guidelines and cognitive research. Since the guideline literature is already compiled, this approach permits more rapid construction of linkages. Sidney Smith has been working on a HyperText interface to his 1986 HCI guidelines that would permit a good foundation. The principal problems with this approach are the focus of the Smith and Mosier guidelines (primarily system operation and usability) and the relative age of the underlying cognitive and HCI research. Nevertheless, these facets could be enhanced and updated.

5.2.3 Further Validation of the CSE Design Framework Process and Product Benefits

One case study cannot validate an entire construct. The research presented here suggests that the conscious inclusion of cognitive task requirements in the design of information presentation and interaction routines should enhance human-computer cooperative decision performance. Further exploration is required to assess the predictability of the following development dimensions:

- predictability of HCI designs based on the CSE Design Framework,
- predictability of expected performance improvement, and

- predictability of cost/benefit tradeoffs.

The first issue is critical for both the validation and the iterative improvement of the CSE Design Framework. In essence, it constitutes the experimental construct of repeatability. The CSE Design Framework will have to be clarified and the processes stabilized to provide a repeatable process. At this point, comparative studies may be conducted to verify that the average designer can use the approach described to produce HCI designs that routinely meet or exceed expectations. The CSE Design Practitioner's Handbook presented in Appendix A represents the first step in packaging the CSE Design Framework for comparative investigation with designers. Until the predictability is verified, the performance improvements and the cost/benefit studies remain speculative.

This chapter summarized some of the issues raised during development and evaluation of the CSE Design Framework. Preliminary findings appear to support both the feasibility and utility of integrating cognitive task requirements into the process of developing HCI designs for decision aiding. Several options were proposed for reinforcing the framework with methods for better integrating the informal conceptual models into requirements specification and design. Suggestions were also made for improving the links between design goals and HCI guidelines. Finally, three key areas were identified as critical to the validation of the CSE Design Framework. These included the predictability of HCI design products, expected performance improvement, and cost/benefit ratio. The proposed enhancement and validation studies represent a wealth of research opportunities with potential value to both the theory and practice of HCI design.

6. References

Adelman, L., M. Tolcott, and T. Bresnick. (1993). "Examining the Effect of Information Order on Expert Judgment." *Organizational Behavior and Human Processes*, 56, 348-369.

Aiken, P.H. and L. S. Ehrhart. (1991). "Making Information in Computerized Databases More Accessible: A Cognitive Engineering Case Study." In *Proc. 29th Annual Technical Symposium. Advanced Information Interfaces: Making Data Accessible, June 20, 1991.* Washington, DC: ACM, 1991, 101-110.

Allen, R. B. (1982). "Cognitive Factors in Human Interaction with Computers." In A. Badre and B. Shneiderman (Eds.), *Directions in Human/Computer Interaction*. Norwood, NJ: Ablex.

American Heritage Dictionary, Second College Edition. (1985). Boston: Houghton Mifflin, 1985.

Anderson, J. R. (1983). *The Architecture of Cognition.* Cambridge, MA: Harvard University Press.

Andriole, S. J. (1987). "Storyboard Prototyping for Requirements Verification," *Large Scale Systems*, 12, 1987, 231-247.

________. (1989). *Handbook of Decision Support Systems.* Blue Ridge Summit, PA: Tab Books, Inc.

________. (1990). *Information System Design Principles for the 90s: Getting it Right!* Fairfax, VA: AFCEA International Press.

Andriole, S. J. & L. Adelman. (1989). *Cognitive Engineering of Advanced Information Technology for Air Force Systems Design & Development: Prototypes for Air Defense Intelligence & Operations.* Marshall, VA: International Information Systems, Inc.

Andriole, S.J. and L.S. Ehrhart. (1990). *Intelligent Decision Support for Electronic Warfare: The IDSSEW Prototype.* Fairfax, VA: Center for Excellence in Command, Control, Communications and Intelligence, George Mason University, September 1990.

Arthur, L. J. (1992). *Rapid Evolutionary Development: Requirements, Prototyping & Software Creation.* New York: John Wiley, 1992.

Badre, A. N. (1992). "Selecting and Representing Information Structures for Visual Presentation," *IEEE Transactions on Systems, Man, and Cybernetics*, 12(4), 495-504.

Bailey, R. W. (1989). *Human Performance Engineering: Using Human Factors / Ergonomics to Achieve Computer System Usability*, 2nd Ed. Englewood Cliffs, NJ: Prentice-Hall.

Bainbridge, L. (1987). "The Ironies of Automation." In J. Rasmussen, K. Duncan & J. Leplat (Eds.), *New Technology and Human Error*. London: Wiley.

________. (1988). "Types of Representation." In L. P. Goodstein, H. B. Andersen, and S. E. Olsen (Eds.), *Tasks, Errors, and Mental Models*. London: Taylor & Francis.

Ball, R. G. and G. Ord. (1983). "Interactive Conflict Resolution in Air Traffic Control." In M. J. Coombs and J. L. Alty (Eds.), *Designing for Human-Computer Communication*. London: Academic Press.

Basili, V. & D. Weiss. (1981). "Evaluation of a Software Requirements Document by Analysis of Change Data." In *Fifth IEEE International Conference on Software Engineering*. Washington, DC: Computer Society Press of the IEEE, pp. 314-323.

Battilega, J. A. and J. K. Grange, (Eds.). (1980). *The Military Applications of Modeling*. Wright-Patterson AFB, Ohio: Air Force Institute of Technology Press, 1980.

Beach, L. R. (1992). "Epistemic Strategies: Causal Thinking in Expert and Nonexpert Judgment." In G. Wright and F. Bolger (Eds.), *Expertise and Decision Support*. New York: Plenum Press.

Bennett, K. B. and J. M. Flach. (1992). "Graphical Displays: Implications for Divided Attention, Focused Attention, and Problem Solving." *Human Factors*, 34(5), 513-533.

Bennett, K. B., M. L. Toms, and D. D. Woods. (1993). "Emergent Features and Graphical Elements: Designing More Effective Configural Displays." *Human Factors*, 35(1), 71-97.

Berliner, D. C., D. Angell, and J. Shearer. (1964). "Behaviors, Measures and Instruments for Performance Evaluation in Simulated Environments." In *Proc. of the Symposium and Workshop on the Quantification of Human Performance*. University of New Mexico, Albuquerque, NM.

Bersoff, E. H. (1984). "Elements of Software Configuration Management," *IEEE Transactions on Software Engineering*, Vol. SE-10, No. 1, January 1984, 79-87.

Boar, B. H. (1984). *Application Prototyping: A Requirements Strategy for the 80's*. New York: Wiley-Interscience.

Boehm, B. W. (1975). "Some Experience with Automated Aids to the Design of Large-Scale Reliable Software." *IEEE Transactions on Software Engineering*, 11(8), March 1975, 125-133.

________. (1976). "Software Engineering," *IEEE Transactions on Computers*, 25(12), December 1976, 1226-1241.

________. (1981). *Software Engineering Economics*. Englewood Cliffs, NJ: Prentice-Hall, 1981.

________. (1987). "Improving Software Productivity," *IEEE Computer*, September 1987, 43-57.

Bott, R. (1979) "A Study of Complex Learning: Theory and Methodology," *CHIP Report 82*, La Jolla, CA: Center for Human Information Processing, University of California at San Diego.

Brehmer, B. (1987). "Models of Diagnostic Judgment." In J. Rasmussen, K. Duncan, and J. Leplat (Eds.), *New Technology and Human Error*. New York: Wiley, 1987.

Brooke, J.B. and K.D. Duncan. (1981). "Effects of System Display Format on Performance in a Fault Location Task," *Ergonomics*, 1981, 24, 175-189.

Brooks, F. P. (1988). "Grasping Reality Through Illusion — Interactive Graphics Serving Science," *Proceedings: ACM/SIGCHI Conference on Human Factors in Computing Systems*, May 1988, pp. 1-10.

Bruggere, T. (1979). "Software Engineering Management, Personnel, and Methodology. In *Fourth IEEE International Conference on Software Engineering*, Washington, DC.: IEEE Computer Society Press.

Byrd, T. A., K. L. Cossick, and R. W. Zmud. (1992). "A Synthesis of Research on Requirements Analysis and Knowledge Acquisition Techniques." *MIS Quarterly*, March 1992, 117-138.

Card, S. K., T. P. Moran, and A. Newell. (1983). *The Psychology of Human-Computer Interaction*. Hillsdale, NJ: Lawrence Erlbaum.

Carroll, J. M. and R. L. Campbell. (1988). *Artifacts as Psychological Theories: The Case of Human-Computer Interaction*. [Technical Report RC 13454 (#60225)]. Yorktown Heights, NY: IBM Research Division, T. J. Watson Research Center.

Carroll, J. M., R. L. Mack and W. A. Kellogg. (1988). "Interface Metaphors and User Interface Design." In M. Helander (Ed.) *Handbook of Human-Computer Interaction*. New York: Elsevier.

Carroll, J. M. and J. R. Olson. (1988). "Mental Models in Human-Computer Interaction." In M. Helander (Ed.), *Handbook of Human-Computer Interaction*. Amsterdam: Elsevier.

Carswell, C. M. and C. D. Wickens. (1987). "Information Integration and the Object Display." *Ergonomics*, 30, 511-527.

Castellan, N. J., Jr. (Ed.) (1993) *Individual and Group Decision Making.* Hillsdale, NJ: Lawrence Erlbaum.

Chambers, N. and H. Nagel. (1985). "Computer-Aided Pilots." *Communications of the ACM*, November.

Chapanis, A. (1959). *Research Techniques in Human Engineering*. Baltimore, MD: Johns Hopkins Press.

Cohen, M. S., B. B. Thompson, and J. O. Chinnis, Jr. (1985). *Design Principles for Personalized Decision Aiding: an Application to Tactical Air Force Route Planning*. RADC-TR-85-140. Final Technical Report. McLean, VA: Decision Science Consortium, Inc., July.

Connell, J. L. and L. Shafter. (1989). *Structured Rapid Prototyping: An Evolutionary Approach to Software Development.* Englewood Cliffs, NJ: Prentice Hall.

Coury, B. G. and M. D. Boulette. (1992). "Time Stress and the Processing of Visual Displays." *Human Factors*, 34(6), 707-725.

Daly, E. (1977). "Management of Software Development." *IEEE Transactions on Software Engineering*, 3(3), May, 229-242.

Davis, A. M. (1993). *Software Requirements: Objects, Functions, and States.* Revised Ed. Englewood Cliffs, NJ: Prentice-Hall.

Dept. of Defense (DOD). (1992). *HCI Style Guide Version 2.0. Final Draft.* Washington, D.C.: Department of Defense, September.

diSessa, A. A. (1988). "Knowledge in Pieces." In G. Forman and P. B. Pufall (Eds.), *Constructivism in the Computer Age.* Hillsdale, NJ: Erlbaum.

Dörner, D. (1987). "On the Difficulties People Have in Dealing with Complexity." In J. Rasmussen, K. Duncan and J. Leplat (Eds.), *New Technology and Human Error*. New York: John Wiley, 1987.

Dreyfus, J. L. and S. E. Dreyfus. (1986). *Mind Over Machine: The Power of Human Intuition and Expertise in the Era of the Computer.* New York: Free Press.

Duffy, L. (1993). "Team Decision-Making and Technology." In N. J. Castellan, Jr. (Ed.), *Individual and Group Decision Making.* Hillsdale, NJ: Erlbaum.

Edgell, S. E., R. D. Bright, P. C. Ng, T. K. Noonan, and L. A. Ford (1992). "The Effects of Representation on the Processing of Probabilistic Information." In B. Burns (Ed.), *Percepts, Concepts and Categories: The Representation and Processing of Information*. Amsterdam: Elsevier Science Publishers.

Edland, A & O. Svenson. (1993). "Judgment and Decision Making under Time Pressure: Studies and Findings." In O. Svenson & A. J. Maule (Eds.), *Time*

Pressure and Stress in Human Judgment and Decision Making. New York: Plenum Press, 27-40.

Ehrhart, L.S. (1990). "New Approaches to Human-Computer Interaction Research and Design for Decision Aiding Systems." In *Proc. IEEE International Intelligent Control Symposium,* Philadelphia, PA, 5-7 September.

_______. (1993a). "Integrating Multimedia Information for Theater-Level Command and Control." Final Report. Air Force Office of Scientific Research, Research Development Laboratories, Summer Research Program. Washington, DC: AFOSR. September 1993.

_______. (1993b). *Tasks, Models & Methods: System-Level Evaluation of Human-Computer Interaction (HCI) for Command & Control Decision Support.* (R&D Interim Technical Report #MISE-3-1-3. Submitted to Rome Laboratory/C^3AA under Contract #F30602-92-C-0119). Fairfax, VA: C^3I Center, George Mason University, 1 March 1993.

Ehrhart, L.S. and P.H. Aiken. (1991). "Cognitive Engineering for Intelligent Control System Design: Preserving User Models in Requirements Analysis." In *Proc. of the 1991 American Control Conference. Boston, MA, 26-28 June, 1991.* Evanston, IL: American Automatic Control Council.

Elm, W. C. & D. D. Woods. (1985). "Getting Lost: A Case Study in Interface Design." In *Proceedings of the Human Factors Society,* 29th Annual Meeting.

Fagan, M. (1974). *Design and Code Inspections and Process Control in the Development of Programs.* IBM Report IBM-SDD-TR-21-572. December.

Fleishman, E. A. & M. K. Quaintance. (1984). *Taxonomies of Human Performance: The Description of Human Tasks.* New York: Academic Press.

French, W. L. and C. H. Bell, Jr. (1973). *Organization Development: Behavioral Science Interventions for Organization Improvement.* Englewood Cliffs, NJ: Prentice-Hall.

Gardiner, M. M. & B. Christie. (1987). *Applying Cognitive Psychology to User-Interface Design.* New York: John Wiley & Sons.

Gentner, D. and D. R. Gentner. (1983). "Flowing Waters or Teeming Crowds: Mental Models of Electricity." In D. Gentner and A. L. Stevens (Eds.), *Mental Models.* Hillsdale, NJ: Lawrence Erlbaum.

Gilchrist, A. (1990). " Perception of Surface Blacks and Whites." In I. Rock (Ed.), *The Perceptual World.* New York: W. H. Freeman.

Goldsmith, T.E. and R. W. Schvaneveldt. (1984). "Facilitating Multiple-Cue Judgements with Integral Displays." In J. C. Thomas & M. L. Schneider (Eds.), *Human Factors in Computing Systems.* Norwood, NJ: Ablex.

Gowan, D. B. & J. D. Novak. (1984). *Learning to Learn.* New York: Cambridge University Press.

Green, T. R. G. (1990). "Limited Theories as a Framework for Human-Computer Interaction." In D. Ackerman and M. J. Tauber (Eds.), *Mental Models and Human-Computer Interaction I.* Amsterdam: Elsevier.

Hammond, K. R. (1993). "Naturalistic Decision Making From a Brunswikian Viewpoint: Its Past, Present, Future." In G. A. Klein, J. Orasanu, R. Calderwood, and C. E. Zsambok (Eds.), *Decision Making in Action: Models and Methods*. Norwood, NJ: Ablex.

Harris, J. F. (1994). "Downing of Copters Blamed on Blunders." *Washington Post,* 117(221), July 14: A1, A10.

Harrison, T. M. (1985). "Communication and Participative Decision Making : An Exploratory Study." *Personnel Psychology*, Spring, 38(1), p. 93-116.

Hart, S. G. & L. Staveland. (1988). "Development of NASA-TLX (Task Load Index): Results of Empirical and Theoretical Research. In P. A. Hancock and N. Meshkati, *Human Mental Workload.* New York: North-Holland, 139-183.

Helmreich, R. L. (1988). *Testimony before the U.S. House of Representatives, Committee on Armed Services on the Subject of the USS VINCENNES Downing of Iranian Air Flight 655*. October 6, 1988.

Hermann, C. F. (1972). "Threat, Time, and Surprise: A Simulation of International Crisis." In Hermann, Charles F., (Ed.), *International Crises: Insights from Behavioral Research.* New York: Free Press.

Hill, S., H. P. Iavecchia, J. C. Byers, A. C. Bittner, A. L. Zaklad, R. E. Christ. (1992). "Comparison for Four Subjective Workload Rating Scales." *Human Factors*, 1992, 34(4), 429-439.

Hoc, J.-M. (1989). "Strategies in Controlling a Continuous Process with Long Response Latencies: Need for Computer Support to Diagnosis." *International Journal of Man-Machine Studies*, 30, 47-67.

Hoeber, F. P. (1981). *Military Applications of Modeling; Selected Case Studies*. New York: Gordon & Breach Science Publishers.

Hogarth, R. (1987). Judgement and Choice: the Psychology of Decision, 2nd Edition. New York: John Wiley.

Irving, S., P. Polson, and J. E. Irving. (1994). "A GOMS Analysis of the Advanced Automated Cockpit." In *Proceedings of CHI, 1994* (Boston, MA, April 24-28, 1994). New York: Association of Computing Machinery.

Janis, I. L. (1972). *Victims of Groupthink.* Boston: Houghton Mifflin.

_______. (1989). *Crucial Decisions: Leadership in Policy Making and Crisis Management*. New York: Free Press.

Johnson, P., D. Diaper, and J. Long. (1984). "Tasks, Skills and Knowledge: Task Analysis for Knowledge Based Descriptions." In *Interact '84*, Vol. 1. London: IFIP.

Johnson-Laird, P. N. (1983). *Mental Models*. Cambridge, UK: Cambridge University Press.

Kaempf, G. L., S. P. Wolf, M. L. Thordsen, & G. A. Klein. (1992). *Decision Making in the AEGIS Combat Information Center.* [Technical Report prepared for Naval Command, Control and Ocean Surveillance Center, San Diego, CA (Contract N66001-90-C-6023). Fairborn, OH: Klein Associates, Inc.

Kahan, J. P., D. R. Worley, and C. Stasz. (1989). *Understanding Commander's Information Needs*. [Technical Report R-3761-A]. Santa Monica, CA: The RAND Corporation.

Kantowitz, B. H. (1987). "Mental Workload." In P. A. Hancock (Ed.), *Human Factors Psychology*. Amsterdam: Elsevier Science Publishers, 81-121.

Kieras, D. E. (1988). "Towards a Practical GOMS Model Methodology for User Interface Design." In M. Helander (Ed.), *Handbook of Human-Computer Interaction*. Amsterdam: Elsevier.

_______. (1992). "Diagrammatic Displays for Engineered Systems: Effects on Human Performance in Interacting with Malfunctioning Systems." *International Journal of Man-Machine Studies*, 36, 861-895.

Kieras, D. E and S. Bovair. (1986). "The Acquisition of Procedures from Text: A Production-System Analysis of Transfer of Training." *Journal of Memory and Language*, 25, 507-524.

Kieras, D. E. and P. G. Polson. (1985). "An Approach to the Formal Analysis of User Complexity," *International Journal of Man-Machine Studies*, 1985, 22, 365-394.

Klapp, S. T. (1987). "Short-Term Memory Limits in Human Performance." In P. A. Hancock (Ed.), *Human Factors Psychology*. Amsterdam: North-Holland.

Klein, G. A. (1989a). "Do Decision Biases Explain Too Much?" *Human Factors Society Bulletin*, 35(5), 1-3.

_______. (1989b). "Recognition-Primed Decisions." In W. B. Rouse (Ed.), *Advances in Man-Machine Systems Research, Vol. 5.* Greenwich, CT: JAI.

_______. (1993a). "A Recognition-Primed Decision (RPD) Model of Rapid Decision Making." In G. A. Klein, J. Orasanu, R. Calderwood, and C. E. Zsambok (Eds.), *Decision Making in Action: Models and Methods*. Norwood, NJ: Ablex.

________. (1993b). *Naturalistic Decision Making: Implications for Design*. Technical Report: CSERIAC SOAR 93-1. Wright-Patterson AFB, OH: Crew System Ergonomics, Information Analysis Center, April.

Klein, G. A., R. Calderwood & D. MacGregor. (1989). "Critical Decision Method for Eliciting Knowledge," *IEEE Transactions on Systems, Man and Cybernetics*, 19(3), 462-472.

Klein, G. A., J. Orasanu, R. Calderwood, and C. E. Zsambok, (Eds.). (1993). *Decision Making in Action: Models and Methods.* Norwood, NJ: Ablex.

Kleinmuntz, D. N. (1987). "Human Decision Processes: Heuristics and Task Structure." In P. A. Hancock (Ed.), *Human Factors Psychology*. Amsterdam: North-Holland.

Klinger, D. W., S. J. Andriole, L. G. Militello, L. Adelman, G. Klein. (1993). *Designing for Performance: A Cognitive Systems Engineering Approach to Modifying an AWACS Human-Computer Interface.* Final Technical Report. Contract No. F33615-90-C-0533. Fairborn, OH: Klein Associates, Inc., March.

Leddo, J. M., J. O. Chinnis, M. S. Cohen, and F. M. Martin. (1989). "Tactical Decision Making Under Uncertainty and Time Stress." In *Proceedings of the 1989 Symposium on Command and Control Research*, June 27-29, 1989, National Defense University, Ft. Lesley J. McNair, Washington, DC. McLean, VA: SAIC, December.

Lehner, P. E. and D. A. Zirk. (1987). "Cognitive Factors in User/Expert-System Interaction." *Human Factors*, 29(1), 97-109.

Leiser, B. (1992). "The Presence Phenomenon and Other Problems of Applying Mental Models to User Interface Design and Evaluation." In Y. Rogers, A. Rutherford & P. A. Bibby (Eds.), *Models in the Mind: Theory, Perspective and Application*. New York: Academic Press.

Levis, A. H., N. Moray, and B. Hu. (1994). "Task Decomposition and Allocation Problems and Discrete Event Systems." *Automatica*, 30(2).

Lockhead, G. R. (1992). "On Identifying Things: A Case for Context." In B. Burns (Ed.), *Percepts, Concepts and Categories: The Representation and Processing of Information*. Amsterdam: Elsevier Science Publishers.

Loftus, G.R. and Mackworth, N.H. (1978). "Cognitive Determinants of Fixation Location During Picture Viewing." *Journal of Experimental Psychology: Human Perception and Performance*. 4, 565-572.

MacMillan, J. & E. B. Entin. (1991). "Decision-Oriented Display Design and Evaluation." In *Proceedings of the 1991Symposium on Command and Control Research*, June 24-25, 1991, National Defense University, Ft. Lesley J. McNair, Washington, DC. McLean, VA: SAIC, December.

Mantei, M. M. and T. J. Teorey. (1988). "Cost/Benefit Analysis for Incorporating Human Factors in the Software Lifecycle." *Communications of the ACM*, 31(4), April, 428-439.

Maule, A. J. & O. Svenson. (1993). "Theoretical and Empirical Approaches to Behavioral Decision Making and Their Relation to Time Constraints." In O. Svenson and A. J. Maule (Eds.), *Time Pressure and Stress in Human Judgment and Decision Making*. New York: Plenum Press, 3-25.

Mayer, R.E. (1979) "Can Advance Organizers Influence Meaningful Learning?" *Review of Educational Research*, 49, 371-383.

Mayer, R.E. and B. K. Bromage. (1980). "Different Recall Protocols for Technical Texts Due to Advance Organizers," *Journal of Educational Psychology*, 72, 209-225.

McCormick, E. J. (1976). "Job and Task Analysis." In M.D. Dunnette, (Ed.), *Handbook of Industrial and Organizational Psychology*. Chicago: Rand McNally.

McFarren, M. R. (1987). *Using Concept Mapping to Define Problems and Identify Key Kernels During the Development of a Decision Support System.* Master Thesis. Wright Patterson AFB, OH: Air Force Institute of Technology, School of Engineering.

Meister, D. (1981). *Behavioral Research and Government Policy: Civilian and Military R & D.* New York: Pergamon Press.

_______. (1985). *Behavioral Analysis and Measurement Methods.* New York: John Wiley.

_______. (1991). *Psychology of System Design*. Amsterdam: Elsevier.

Metersky, M. L. (1993). "A Decision-Oriented Approach to System Design and Development." *IEEE Transactions on Systems, Man and Cybernetics*, 23(4),July/August, 1024-1037.

Montazemi, A. R. and D. W. Conrath. (1986). "The Use of Cognitive Mapping for Information Requirements Analysis," *MIS Quarterly*, 10(1), March.

Nielsen, J. (1993). *Usability Engineering*. New York: Academic Press.

Nielsen, J. and R. Molich. (1990). "Heuristic Evaluation of User Interfaces." *Proc. of ACM CHI '90 Conference.* (Seattle, WA, 1-5 April 1990), 249-256.

Nielsen, J. and V. L. Phillips. (1993). "Estimating the Relative Usability of Two Interfaces: Heuristic, Formal, and Empirical Methods Compared." In Proceedings of

Norman, D. A. "Some Observations on Mental Models." In D. Gentner and A. L. Stevens (Eds.), *Mental Models*. Hillsdale, NJ: Lawrence Erlbaum, 1983.

_______. (1986). "Cognitive Engineering." In D. A. Norman and S. W. Draper (Eds.), *User-Centered System Design: New Perspectives on Human-Computer Interaction.* Hillsdale, NJ: Lawrence Erlbaum.

Norman, D. A. and S. W. Draper (Eds.). (1986). *User-Centered System Design: New Perspectives on Human-Computer Interaction.* Hillsdale, NJ: Lawrence Erlbaum.

Norman, K. L., L. J. Weldon, and B. Shneiderman. (1986). "Cognitive Layouts of Windows and Multiple Screens for User Interfaces," *International Journal of Man-Machine Studies*, 25, 229-248.

Nygren, T. E. (1991). "Psychometric Properties of Subjective Workload Measurement Techniques: Implications for Their Use in the Assessment of Perceived Mental Workload." *Human Factors*, 1991, 33(1), 17-33.

O'Malley, C. & S. Draper. "Representation and Interaction: Are Mental Models All in the Mind?" In Y. Rogers, A. Rutherford & P. A. Bibby (Eds.), *Models in the Mind: Theory, Perspective and Application*. New York: Academic Press, 1992.

Orasanu, J. and E. Salas. (1993). "Team Decisionmaking in Complex Environments." In G. A. Klein, J. Orasanu, R. Calderwood, and C. E. Zsambok (Eds.), *Decision Making in Action: Models and Methods.* Norwood, NJ: Ablex.

Patrick, J. (1987). "Information at the Human-Machine Interface." In J. Rasmussen, K. Duncan & J. Leplat (Eds.), *New Technology and Human Error.* London: Wiley.

Payne, J. W., J. R. Bettman, and E. J. Johnson. (1993). "The Use of Multiple Strategies in Judgment and Choice." In N. J. Castellan, Jr. (Ed.), *Individual and Group Decision Making.* Hillsdale, NJ: Lawrence Erlbaum.

Payne, S. J. (1984). "Task-Action Grammars." In *Interact '84*, Vol. 1. London: IFIP.

Perdu, D. and A. H. Levis. (1993). "Requirements Determination Using the Cube Tool Methodology and Petri Nets." *IEEE Transactions on Systems, Man, and Cybernetics*, 23(5), September.

Pew, R. W. (1988). *Testimony before the U.S. House of Representatives, Committee on Armed Services on the Subject of the USS VINCENNES Downing of Iranian Air Flight 655.* October 6, 1988.

Raphael, T. D. (1991). *Cognitive Flexibility and Innovation in the Problem Definition Process for Tactical Command and Control*. [D-39W-91 (Revised). Prepared for Logistics Research Group, Air Force Human Relations Laboratory, Wright-Patterson AFB, OH 45433-6573.] Falls Church, VA: Mystech Associates, May.

Rasmussen, J. (1990). "Mental Models and the Control of Action in Complex Environments." In D. Ackerman and M. J. Tauber (Eds.), *Mental Models and Human-Computer Interaction I*. Amsterdam: Elsevier (North-Holland).

_______. (1986). *Information Processing and Human-Machine Interaction: an Approach to Cognitive Engineering*. New York: North-Holland, 1986.

_______. (1993). "Deciding and Doing: Decision Making in Natural Contexts." In G. A. Klein, J. Orasanu, R. Calderwood, and C. E. Zsambok (Eds.), *Decision Making in Action: Models and Methods*. Norwood, NJ: Ablex.

Rasmussen, J. & L. P. Goodstein. (1988). "Information Technology and Work," in Martin Helander (Ed.), *Handbook of Human-Computer Interaction*. Amsterdam: Elsevier.

Rasmussen, J. and A. M. Pejtersen. (1993). *Mohawc Taxonomy Implications for Design and Evaluation.* Technical Report Risø-M-673(EN). Roskilde, Denmark: Risø National Laboratory, January.

Rasmussen, J. and K. J. Vicente. (1989). "Coping with Human Errors through System Design: Implications for Ecological Interface Design." *International Journal of Man-Machine Studies*, 31, 517-534.

Rasmussen, J., A. M. Pejtersen, and K. Schmidt. (1990). *Taxonomy for Cognitive Work Analysis.* Technical Report Risø-M-2871. Roskilde, Denmark: Risø National Laboratory, September.

Rasmussen, J., A. M. Pejtersen, and L. P. Goodstein. (in press). *Cognitive Systems Engineering.* New York: John Wiley.

Rastegary, H. and F. J. Landy. (1993). "The Interactions Among Time Urgency, Uncertainty, and Time Pressure." In O. Svenson and A. J. Maule (Eds.), *Time Pressure and Stress in Human Judgment and Decision Making*. New York: Plenum Press, 217-240.

Ravden, S. J. and G. I. Johnson. (1989). *Evaluating Usability of Human-Computer Interfaces: a Practical Method.* Chichester, UK: Ellis Horwood.

Robbins, Stephen P. (1990). *Organizational Theory: Structure, Design and Applications.* Englewood Cliffs, NJ: Prentice-Hall.

Rogers, Y. (1992). "Mental Models and Complex Tasks." In Y. Rogers, A. Rutherford & P. A. Bibby (Eds.), *Models in the Mind: Theory, Perspective and Application*. New York: Academic Press.

Roth, E.M., K.B. Bennett, and D.D. Woods. (1988). "Human Interaction with an "Intelligent" Machine." In E. Hollnagel, G. Mancini, and D. D. Woods (Eds.), *Cognitive Engineering in Complex Dynamic Worlds.* London: Academic Press.

Rouse, W. B. (1991). *Design for Success: a Human Centered Approach to Designing Successful Products and Systems*. New York: John Wiley.

Rouse, W. B., J. A. Cannon-Bowers, and E. Salas. (1992). "The Role of Mental Models in Team Performance in Complex Systems." *IEEE Transactions on Systems, Man, and Cybernetics*, .

Salama, A. (1992). "The Use of an Organization's Biography as a Research Method for Investigating Organizational Development." *Management Education & Development*, Autumn, 23(3), 225-233.

Salvendy, G. (1987). "What We Know and What We Should Know About Human-Computer Interaction: Strategies for Research and Development." In G. Salvendy (Ed.), *Cognitive Engineering in the Design of Human-Computer Interaction. Proc. of the Second International Conference on Human-Computer Interaction, Honolulu, Hawaii, August 10-14, 1987, Vol II.* New York: Elsevier,.

Sanderson, P. M., J. M. Flach, M. A. Buttigieg, and E. J. Casey. (1989). "Object Displays Do Not Always Support Better Integrated Task Performance." *Human Factors*, 31(2), 183-198.

Senders, J. W. and N. P. Moray. (1991). *Human Error: Cause, Prediction, and Reduction*. Hillsdale, NJ: Lawrence Erlbaum.

Shanteau, J. (1992). "The Psychology of Experts: An Alternative View." In G. Wright and F. Bolger (Eds.), *Expertise and Decision Support*. New York: Plenum Press.

Sheridan, T. B. and W. R. Ferrell. (1974). *Man-Machine Systems: Information, Control and Decision Models of Human Performance.* Cambridge, MA: The MIT Press.

Shneiderman, B. (1992). *Designing the User Interface: Strategies for Effective Human-Computer Interaction*, 2nd Ed. New York: Addision-Wesley.

Smith, J. B., D. K. Smith, and E. Kupstas. (1993). "Automated Protocol Analysis." *Human-Computer Interaction*, 8, 101-145.

Smith, S. L. & J. N. Mosier. (1986). *Guidelines for Designing User Interface Software*, Report ESD-TR-86-278. Bedford, MA: The MITRE Corporation, August.

Staggers, N. and A. F. Norcio. (1993). "Mental Models: Concepts for Human-Computer Interaction Research." *International Journal of Man-Machine Studies*, 38, 587-605.

Stammers, R. B., M. S. Carey and J. A. Astley. (1990). "Task Analysis," in J. R. Wilson and E. N. Corlett, (Eds.). *Evaluation of Human Work: A Practical Ergonomics Methodology*. New York: Taylor & Francis.

Svenson, O. and A. J. Maule (Eds.). (1993). *Time Pressure and Stress in Human Judgment and Decision Making*. New York: Plenum Press

Tainsh, M. A. (1985). "Job Process Charts and Man-Computer Interaction Within Naval Command Systems." *Ergonomics*, 28, 555-565.

Thayer, T. A. (1975). "Understanding Software Through Analysis of Empirical Data," *TRW Software Series*, TRW-SS-75-04, May.

Treu, S. (1992). "Interface Structures: Conceptual, Logical, and Physical Patterns Applicable to Human-Computer Interaction." *International Journal of Man-Machine Studies*, 37, 565-593.

U.S. Army Research Institute. (1989). *Human Factors Guidelines for Command and Control Systems: Battlefield and Decision Graphics Guidelines. Research Product 89-01.* Fort Leavenworth, KS: U. S. Army Research Institute for the Behavioral and Social Sciences, March.

Vanderhaegen, F., I. Crevits, S. Debernard, P. Millot. (1994). "Human-Machine Cooperation: Toward an Activity Regulation Assistance for Different Air Traffic Control Levels." *International Journal of Human-Computer Interaction*, 6(1), 65-104.

Vicente, K. J. and J. Rasmussen. (1992). "Ecological Interface Design: Theoretical Foundations." *IEEE Transactions on Systems, Man and Cybernteics*, 22(4), July/August, 589-606.

Weick, K. E. and K. H. Roberts. (1993). "Collective Mind in Organizations: Heedful Interrelating on Flight Decks." *Administrative Quarterly*, 38(3), September, 357-381.

Welford, A. T. (1976). *Skilled Performance: Perceptual and Motor Skills*. Glenview, IL: Scott, Foresman & Co.

Wellens, A. R. (1993). "Group Situation Awareness and Distributed Decision Making: From Military to Civilian Applications." In N. J. Castellan, Jr. (Ed.), *Individual and Group Decision Making*. Hillsdale, NJ: Lawrence Erlbaum.

Wickens, C. D. (1987). "Attention." In P. A. Hancock (Ed.), *Human Factors Psychology*. Amsterdam: North-Holland.

Wickens, C. D., A. Stokes, B. Barnett, and F. Hyman. (1993). "The Effects of Stress on Pilot Judgment in a MIDIS Simulator." In O. Svenson and A. J. Maule (Eds.), *Time Pressure and Stress in Human Judgment and Decision Making*. New York: Plenum Press, 271-292.

Williams, K. E., R. E. Reynolds, T. F. Carolan, P. D. Anglin, and L. B. Shrestha. (1989). *An Evaluation of a Methodology for Cognitively Structuring and Adaptively Sequencing Exercise Content for Embedded Training.* Technical

Report TR89-035. Orlando, FL: University of Central Florida, Institute for Simulation and Training, December.

Wilson, J. and D. Rosenberg. (1988). "Rapid Prototyping for User Interface Design." In M. Helander (Ed.), *Handbook of Human-Computer Interaction*. Amsterdam: North-Holland.

Woods, D. D. (1984). "Visual Momentum: A Concept to Improve the Cognitive Coupling of Person and Computer." *International Journal of Man-Machine Studies*, 21, 229-244.

________. (1988). "Coping with Complexity: The Psychology of Human Behavior in Complex Systems." In L. P. Goodstein, H. B. Andersen, and S. E. Olsen (Eds.), *Tasks, Errors, and Mental Models*. London: Taylor & Francis.

________. (1991). "The Cognitive Engineering of Problem Representations." In G. R. S. Weir and J. L. Alty (Eds.), *Human-Computer Interaction and Complex Systems*. London: Academic Press.

Woods, D. D. and E. Hollnagel. (1987). "Mapping Cognitive Demands in Complex Problem-Solving Worlds." *International Journal of Man-Machine Studies*, 26, 257-275.

Woods, D. D. and E. M. Roth. (1988). "Cognitive Systems Engineering." In M. Helander (Ed.), *Handbook of Human-Computer Interaction*. Amsterdam: Elsevier

Wright, P. C. and A. F. Monk. (1989). "Evaluation for Design." In A. Sutcliffe and L. Macaulay (Eds.), *People and Computers V*. Cambridge, UK: Cambridge University Press.

Wright, P. C. and A. F. Monk. (1991). "A Cost-Effective Evaluation Method for Use by Designers." *International Journal of Man-Machine Studies*, 35, 891-912.

Zachary, W. W. (1988). "Decision Support Systems: Designing to Extend the Cognitive Limits." In M. Helander (Ed.), *Handbook of Human-Computer Interaction*. Amsterdam: Elsevier.

Zahniser, R. A. (1993). "Design by Walking Around." *Communications of the ACM*, 36(10), October, 115-123.

Zhang, W.-R., S.-S. Chen, W. Wang, and R. S. King. (1992). "A Cognitive-Map-Based Approach to the Coordination of Distributed Cooperative Agents." *IEEE Transactions on Systems, Man, and Cybernetics*, 22(1), Jan/Feb, 103-113.

Appendix A:

A Cognitive Systems Engineering Design Practitioner's Handbook

A Cognitive Systems Engineering Design Practitioner's Handbook

Cognitive systems engineering (CSE) involves the application of research findings and design experience, drawn from the cognitive science and engineering disciplines. The primary objective of human-computer interaction (HCI) design for decision support systems is ensuring that the human decision-maker gets the *right information* at the *right time* with the *right level of detail*. CSE methods guide the matching of user, task, and organizational/environmental requirements to available tools and techniques for the design of human-machine cooperative decision-making.

In direct contrast to the often noted tendency for "technology push" in advanced systems development, the CSE emphasis on the support needs of the decision-maker represents *requirements-driven* design. The key premise in this framework is the notion that, in addition to the interaction task requirements (IRTs) associated with operating the interface, process improvement hinges upon identifying a more comprehensive set of human cognitive task requirements (CRTs) and successfully translating those requirements into design concepts. The resulting system should demonstrate consistently high human-computer decision task performance as determined by appropriate measures of performance and effectiveness.

This handbook outlines a CSE framework for requirements identification and representation and conceptual design phases of system development to improve human-computer interaction (HCI) designs for decision support. The framework focuses on the application of research and technology in developing a more comprehensive *understanding, representation,* and *translation* of the human decision-maker's information requirements for decision support system design. The design guidance tables contained in the appendices summarize research from software engineering, decision sciences, cognitive psychology and other related fields to assist the designer in defining a more robust set system requirements and guide design tradeoff decisions.

A system design case study in cooperative human-computer decision-making demonstrates the practical implementation of CSE for HCI design, guiding the reader through the application of the guidance tables to a "real world" design problem. The case study also provides the means to evaluate the benefits of this framework for creating HCI designs that improve performance.

The Cognitive Systems Engineering (CSE) Framework for HCI Design and Development

Bersoff [1984] defines *product integrity* as a measure of the extent the delivered product satisfies the real needs and the cost, schedule and performance expectations of the user. The traditional systems engineering model comprises an iterative, multi-phase process to guide designers in developing effective systems. The essential phases include:

1. **Problem definition** - *understanding* problem dimensions to enable problem structuring (*why* the system is needed);
2. **Requirements identification & modeling** - *representing* system response goals to support design specification (*what* is needed);
3. **Design** - *translating* requirements into a functional technological solution (*how* to meet identified needs);
4. **Implementation** - realizing the technological solution; and

5. **Operational testing & evaluation** - verifying and validating system performance against requirements goals and design specifications.[1]

The system development process is, at its core, a problem solving exercise in which much of the solution is suggested by the recognition of the problem. To assist the development of decision-oriented displays, Metersky [1993] proposes an iterative prototyping approach to system design and development that highlights the requirements of the human decision-maker. Andriole [1990] presents the requirements and design prototyping process as a miniature version of the larger system development process. In similar fashion, the CSE prototype design framework proposes an iterative sequence of activities that correlate with traditional systems development phases (Figure A-1). This connection promotes smoother integration of prototyping activities and findings into the overall development effort. The information inputs, sub-tasks and process outputs for each of the six phases are explained below.

Phase One - Defining the Problem

During this phase (Figure A-2) the design team gathers information to understand the functional goals of the system as defined by the sponsoring organization. Information drawn from various organizational documents and discussions with the sponsor help to develop a high-level profile of the system context as defined by:

- **system boundaries** - problem domain, system scope, major subsystems included, conditions of use, etc.;
- **constraints** - pre-defined hardware, software, and communication requirements;

---

[1] Note: Each phase of this process involves internal testing and evaluation to verify that the products developed at that phase meet the stated objectives.

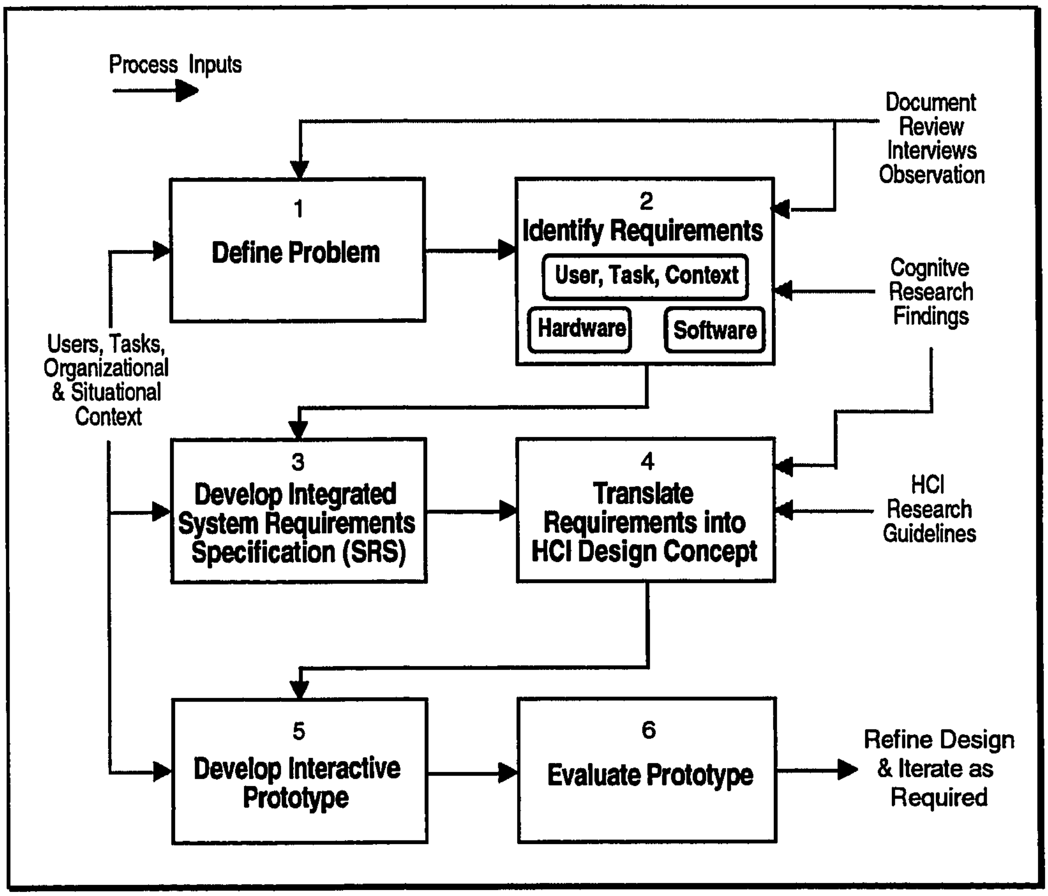


Figure A-1: The CSE Framework for Prototype Design & Development

- **technological opportunities** - potential applications of advanced technologies to provide performance improvement;
- **proposed system inputs** - input sources, control structures, communication modes, etc.;
- **goals & objectives** - high-level functions, organizational goals and missions;
- **technical feasibility** - determination of general feasibility of system goals & development requirements, and

- **proposed system outputs** - high-level performance goals, communication modes, etc.

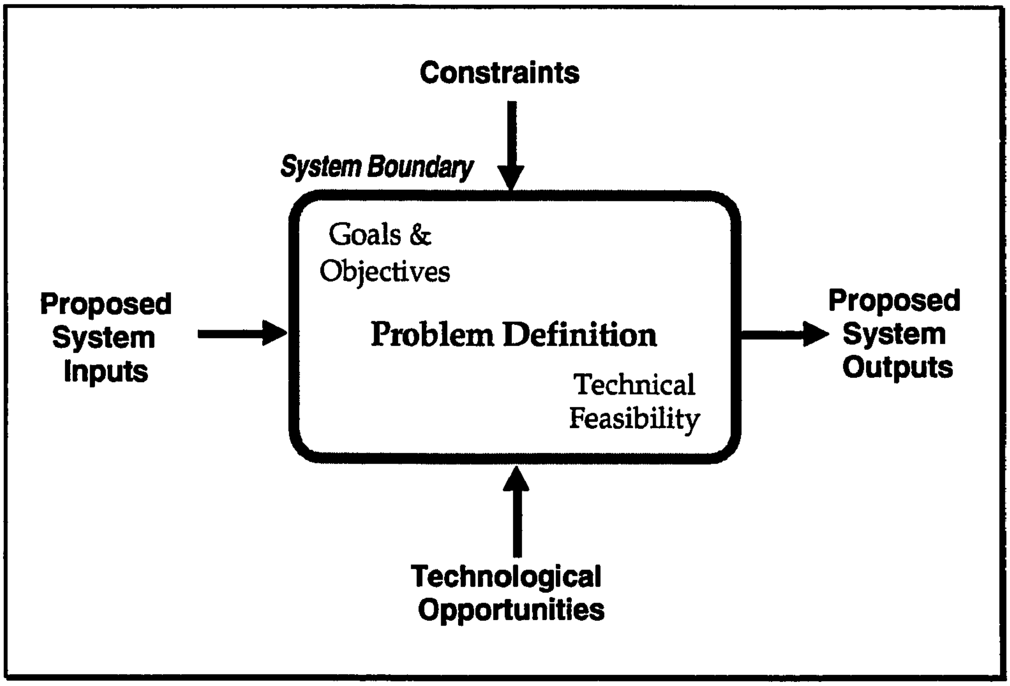


Figure A-2: Issues in Problem Definition Process

The most useful outputs from this phase are preliminary models, such as concept maps and functional decomposition diagrams, defining these central constructs and indicating relationships between them. One of the most difficult aspects of the definition process is the internal (and sometimes external) pressure to "define" in terms of solutions. Jumping to solution thinking during this phase focuses the later requirements analysis exclusively on those problem aspects related to the proposed solution and leads to one of the most common sources of error -- defining the wrong problem and then proceeding to solve it.

Problem definition activities vary widely in the granularity of representation required. The same design may use different modeling methods for different development efforts. For this reason, the CSE framework does not specify or require any particular modeling method; rather it is left to developer to ascertain which methods will best address the issues of interest.

The FLEX Case Study - Defining the Problem

The development of the Force-Level Execution (FLEX) prototype at the Air Force's Rome Laboratory presents an excellent opportunity for applying and evaluating the CSE framework for HCI design. As indicated in Figure A-3, FLEX is intended to support the Combat Operations Division (COD) of the Air Operations Center (AOC) in the execution of the active Air Tasking Order (ATO). The decision environment is complex and dynamic involving a high degree of uncertainty combined with high threat. The duty officers (DOs) in the COD monitor the evolving situation and re-plan the ATO activities to meet changes in goals and/or available resources.[2] Figures C-1 to C-11 in Appendix C present examples of the high-level models of the COD tasks and interactions. Figures G-1 through G-7 in Appendix G present modified versions of the existing FLEX system windows that apply to the Tanker Operations tasks addressed in this case study.

To provide a tractable example, the CSE case study is focusing only on the FLEX re-planning support to the Tanker Duty Officer (TDO). The Tanker Duty Officer (TDO) is responsible for providing air refueling (AR) support to all scheduled missions which require refueling. Re-planning is required when new missions are created, existing missions re-routed, or air refueling resources change. The TDO performs re-planning tasks as indicated by their

---

[2] The Duty Officer in AOC is a decision-maker, thus, in discussions of FLEX the term decision-maker (DM) is used interchangeably with duty officer (DO).

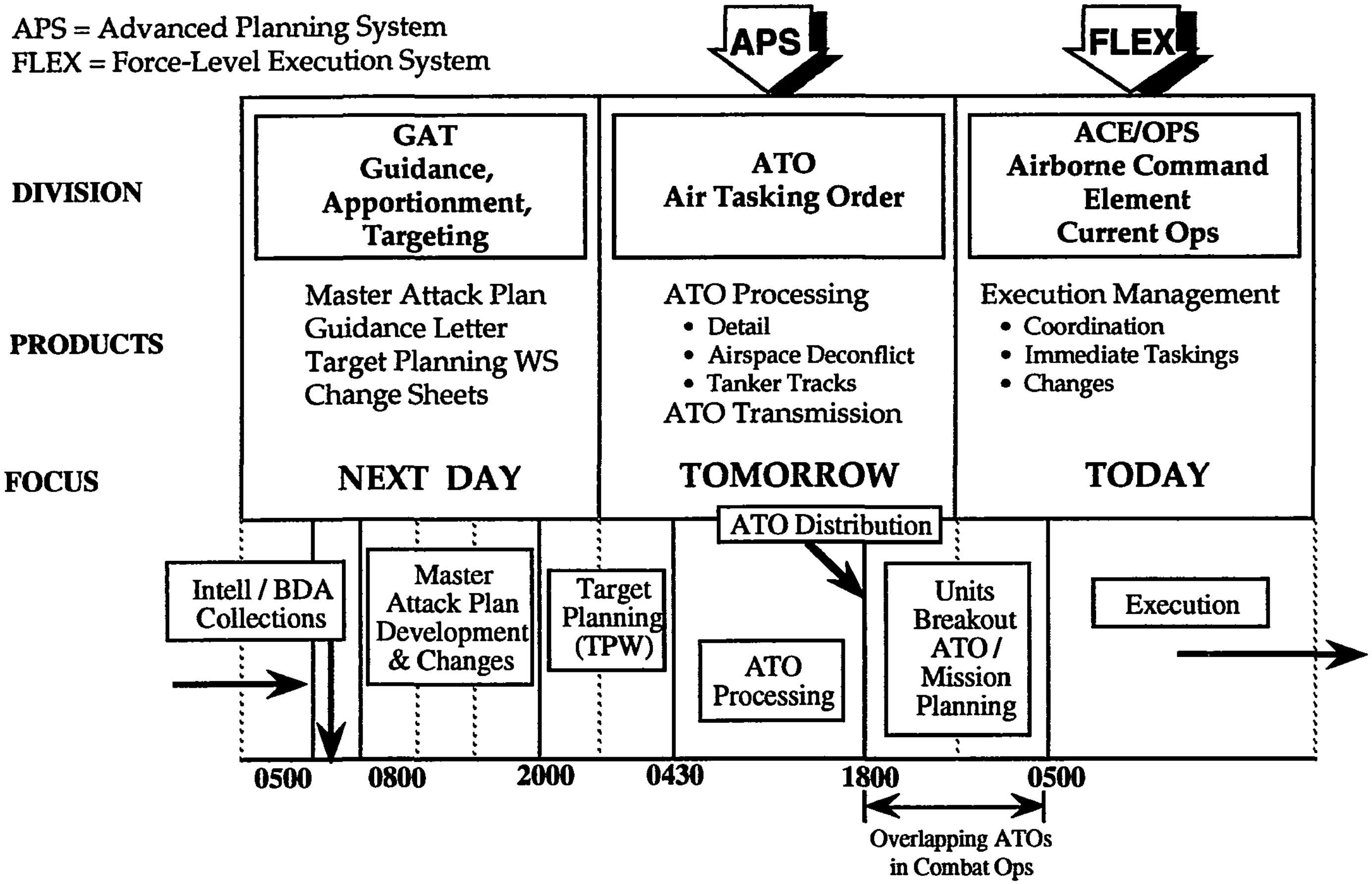


Figure A-3: FLEX & the ATO Timeline

own assessment of the evolving situation and as tasked by other duty officers. Figures B-12 through B-15 in Appendix C present examples of the cognitive maps developed to model TDO tasks and decision variables.

Phase Two - Requirements Identification

During the requirements analysis phase of prototype development, the cognitive task requirements (CTRs) of the user can be identified and defined as part of the normal requirements identification activities. Using the high-level conceptual models from the early problem definition activities and the evolving hardware and software requirements, the HCI designer develops models of information flows, task allocations, and organizational procedures for decision-making. At this point, it is useful to observe the way the organization currently addresses the problem and interview representative users to expand and correct the preliminary functional, procedural, and dependency models. These data activities are part of the *standard* requirements identification and analysis processes. Figure A-4 indicates the interaction of user/organization characteristics and the environmental/situational context in task definition. Included are references to the tables in Appendix B that provide guidance for identifying the key issues in each requirements dimension that impact the overall effectiveness of the cooperative human-machine decision system.

A wealth of requirements modeling methods are available to the designer-analyst. Byrd *et al* [1992] survey requirements analysis and knowledge acquisition techniques that facilitate problem domain understanding in terms of information requirements, process understanding, behavior understanding and problem frame understanding. Multi-perspective context models, such as those described in Davis [1993], assist in creating informal models for review and iteration with the sponsors and operational users.

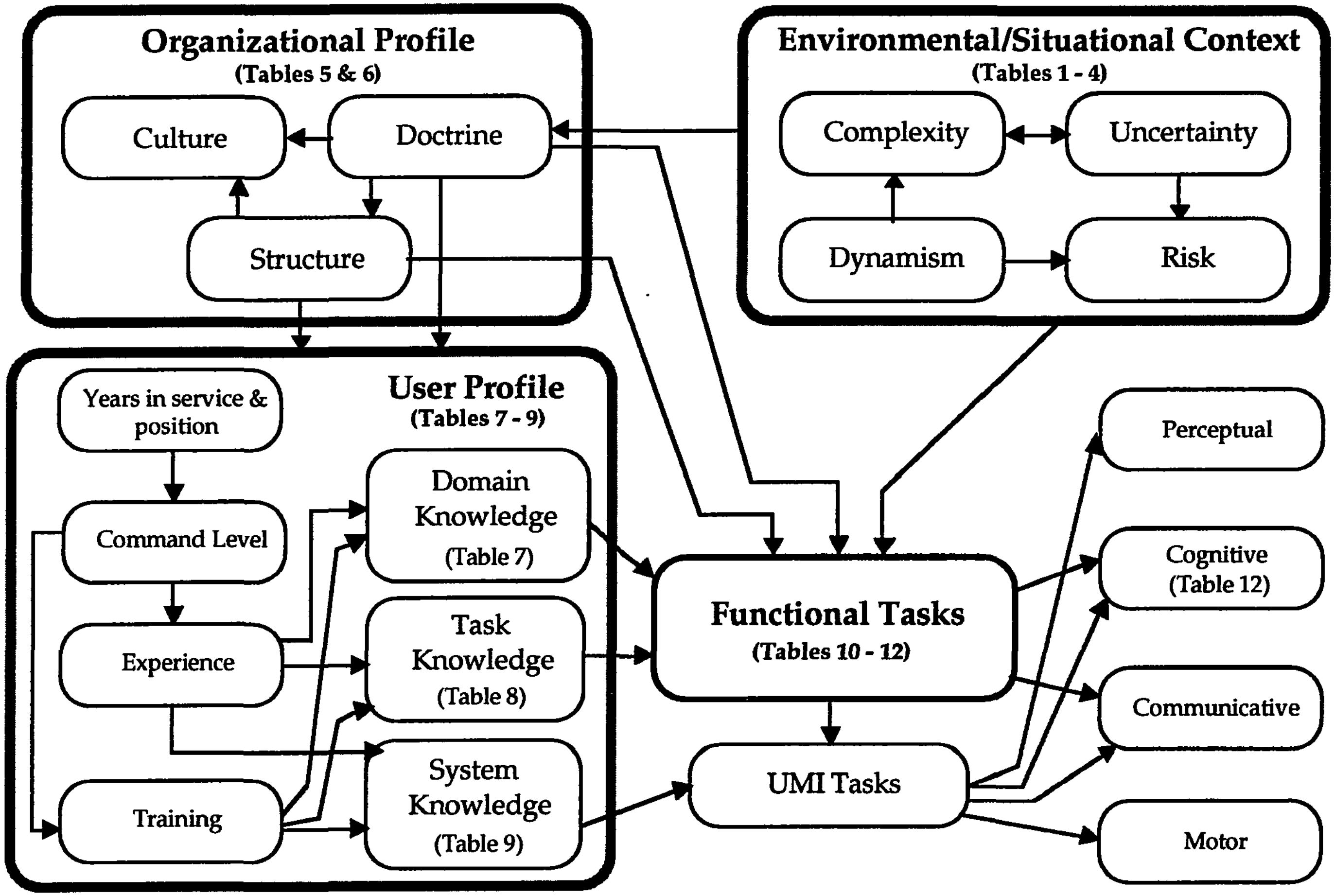


Figure A-4: Requirements Identification

Similarly, Zahniser [1993] describes the creation of *N*-dimensional views of the system developed by cross-functional development teams. The process is designed to foster innovative thinking and bring multi-disciplinary experience to bear on system development problems. The CSE framework complements this effort by including a model of the user's cognitive tasks (as implied by the information flows or prescribed by operational procedures) and analyzing that model with respect to the user's information requirements and the possible sources of cognitive errors.

The HCI design requirements provide a focal point for integrating the information gathered on the users, problem solving tasks, and the decision environment to guide design decisions involving interaction control and focus of attention. These requirements include not only the interaction task requirements (ITRs) that define the operation of the interface, but also the cognitive task requirements (CTRs) that define the supports for human decision task performance. Cognitive task taxonomies, such as those found in Fleishman & Quaintance [1984] and Rasmussen *et al* [1990] can be used as a filter to identify and categorize cognitive tasks. Similarly, Andriole & Adelman [1989] present a taxonomic discussion of human information processing and inferencing tasks with respect to the potential cognitive errors associated with each.

The system designer uses information from the requirements analysis to minimize system interference with cognitive task performance and direct the user's attention to critical information. Particularly in cases where the tasks are complex and must be performed in a dynamic, time-stressed environment, the designer needs this information to determine information representation modes, display formatting, and data presentation rates. Appendix B presents a series of tables which provide a structure for examining each of the major requirements dimensions. Tables B-1 - 4 help to characterize the situational context in terms of the uncertainty, complexity, volatility, and risk of the decision environment.

Table B-5 identifies some of the information of interest at each organizational level and investigation methods for developing the organizational/doctrinal profiles. Tables B-6a-b chart the ways organizations adapt to changing decision requirements, the situational triggers, organizational responses, and implications for system design. The profile of the end-user incorporates models of the user's expected level of expertise in terms of their knowledge of the operational domain (Table B-7), the functional tasks (Table B-8), and the mechanics of operating the system (Table B-9).

Profiling the end-user's functional tasks involves identifying the key variables and the processes by which values for those variables are detected or inferred, interpreted, and combined during decision-making. Tables B-10, B-11, and B-12 provide a structure synthesized from cognitive research to assist the analyst in identifying the relevant variables, characteristic dimensions, and impacts on performance and system design. Table B-10 presents the characteristics and HCI design issues associated with the system inputs, outputs, and feedback that constitute the key decision variables. Tables B-11a-e identify general characteristics associated with the task outputs, required response, procedures and subtasks, stimuli, and feedback. The high-medium-low scales help to mark the boundaries of the continuum for each characteristic and guide the information gathering process. Tables B-12a-d apply the same structure to the HCI design issues impacted by the various phases of decision-making. These tables organize the decision-making tasks using Wohl's (1981) Stimulus-Hypothesis-Option-Response (SHOR) model.

Throughout the task characterization process the designer is mapping the dimensions of task in terms of the potential implications for HCI design issues such as

- task allocation between human users and machine support,

- information presentation and interaction needs,
- error modes, and
- cognitive workload.

Output of this process is formalized in the requirements which form the HCI design goals.

As the development team reviews the context diagrams, functional decomposition diagrams, and straw man storyboards, descriptions of activities can be examined for verbal constructs that indicate human decision-maker actions. For example, in systems where the human decision-maker must *monitor* a situation and *interpret* evolving events, the software designers may view the inputs to the user as updates to a data base. From the user's perspective, however, this requirement has implications not only for interface operation design, but also for the information presentation design. In order to interpret those updates, the changes must not only be visible to the user, but also presented within a meaningful context. Using the concepts of analogical representation and causal reasoning, this context might include some mapping of relationships between key factors, tracing of changes in relevant factors over time, and/or models of a goal state to which certain parameters should conform (Figure A-5).

The FLEX Case Study - Identifying and Modeling the Cognitive Task Requirements

Since the case study was external to the actual FLEX development effort, the CTR identification process began with the examination of system requirements information gathered from a variety of sources including:

- **Document Reviews** - RL development team trip reports, FLEX statement of work, contract developer's system requirements specification (SRS) and system software design documents, written change requests, and a variety of Air Force manuals and support materials on air refueling operations. Appendix D provides a complete list of the documents consulted for requirements identification and modeling.

- **Interviews** - interviews with RL team, the contract development teams, FLEX working group (operational personnel from major commands), and tanker operations personnel from Griffiss AFB's 509th Air Refueling Squadron.
- **Observation** - observation of FLEX working group officers interacting with the three prototype interactions of the FLEX interface.

These materials were used to iteratively refine the models of the air refueling domain, the TDO and the tanker re-planning tasks (Appendix C).

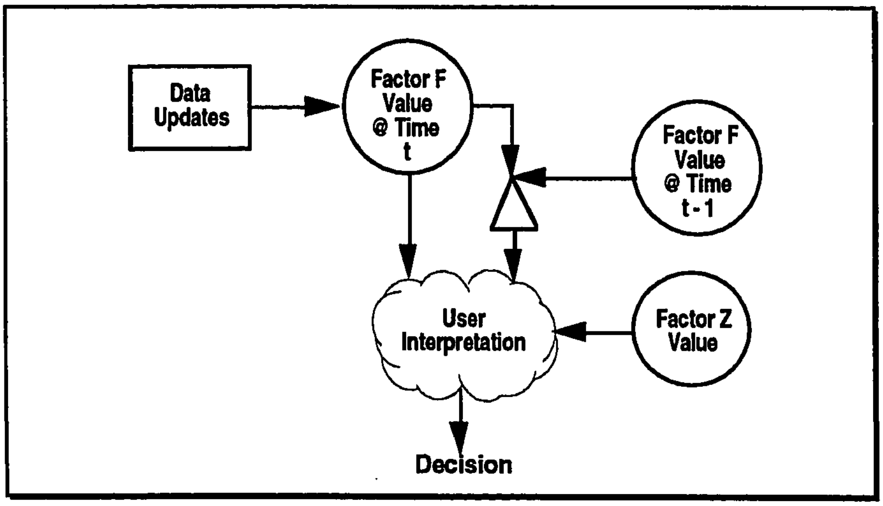


Figure A-5: Model of the User's Information Processing and Inferencing Activities in an Example Decision-making Task

Defining the FLEX Environment/Situational Context

Referencing the tables in Appendix B, the user/organization, task, and environmental/situational models evolved into a set of cognitive task requirements (CTRs) that became the design objectives for the CSE interface prototype. In combat situations, activities in the COD take place in an environment (Table B-1) which ranges from severely stochastic (e.g., the coordination of a complex array of friendly assets) to indeterminate (e.g., mission perturbations caused by an intelligent adversary). As indicated in Table B-2, this situational context has several impacts:

- **Organizational Goals** - Develop means to make most efficient use of resources in a succession of varying short-term situations; rapidly & effectively exploit opportunities; and support maximum flexibility and adaptiveness in novel situations.
- **Potential Errors** - Potential misallocation of resources due to latency between recognition of situation & internal readjustment; achieving flexibility at the cost of control.
- **Information Requirements** - Understanding the problem structure is all important; requires overview displays to relate functional relationships and provide externalized mental models; requires easy access to multiple levels of abstraction to assist adaptive cognitive control requirements.

Decision tasks in the COD range from semi-structured to unstructured due to the high volume of information and potential for "unknown unknowns" (Table B-3).

Profiling the FLEX Organizational/Doctrinal Context

The COD is part of a hierarchical organization which is both vertically and horizontally complex with a moderately-high interdependency between functional units (Table B-4). The control structures in adaptive decision-making organizations shift in response to changes in the decision requirements (Table B-6b). Thus, the general tendency toward the more formal organization evidenced during routine operations shifts during crisis situations to accommodate the requirement for a more flexible response. Table B-6a presents the situational context which triggers shifts in organizational response, the effects of those shifts on decision-making activities, and the design implications for supporting this adaptive environment.

Profiling the Tanker Duty Officer

The profile of the Tanker Duty Officer (TDO) incorporates not only their knowledge of the specific functional tasks assigned to them and their ability to operate the system, but also their understanding of goals and characteristics of the larger domain in which those tasks are performed. Table B-7 presents the defining characteristics, potential errors and system design implications associated with the user's expected level of domain knowledge. The TDO is typically an Air Force major or lieutenant colonel with a moderately high knowledge of the air operations domain acquired through experience, training, and service schools. Many of the errors in situation assessment may be traced to the decision-maker's knowledge of the operational context. Although domain-knowledgeable TDOs may exhibit the ability to intuitively interpret novel situations, they may not be consistent in their combination of situational cues. Situations triggering multiple models may be interpreted based on the more available or vivid model.

The TDO's knowledge of the specific functional tasks assigned them in the COD may also vary depending upon their previous command center experience. Their moderate to high task experience potentially triggers errors associated with the heuristics used to reduce the high workloads during ATO execution (Table B-8). For example, in high information volume situations, the moderately knowledgeable TDO may not have adequate schema to distinguish relevant vs. irrelevant information. They may also erroneously focus on task features that match stored (especially readily available) schema. More experienced TDO's are still vulnerable to a general insensitivity to the potential aggregation of error in the microdecisions performed in multi-stage decision-making. For example, they may tend toward overconfidence in their current decisions and fail to revise when the situation changes. Finally, there is a general tendency for the TDO to think in linear sequences rather than networks of contributing causes and branching consequences of actions.

The TDO's system interaction/operation knowledge will typically be the most variable dimension. In the absence of a protracted war, the majority of the officers assigned to the COD will be casual to competent system users (Table B-9). That is, they will not routinely have to operate the system under the time-critical, high workload conditions which characterize combat operations. Adequate operation of the system during routine or training operations will deteriorate under stress resulting in a variety of errors and an increased level of frustration and confusion.

Profiling the TDO's Functional Tasks

The TDO functional tasks were reviewed using information in these tables and filtering them through the user, organization, and situational context profiles described above. This process identified several key dimensions which defined task performance and error modes, including:

- task complexity and difficulty;
- task performance precision & accuracy requirements;
- input and feedback uncertainty; and
- task workload & potential stress dimensions.

It should be noted that probing task dimensions often triggers further refinement of the other profiles and all of this investigation involved repeated iteration in both top-down and bottom-up analyses.

The FLEX CTRs

Appendix E presents a summary of the issues raised during the CTR identification phase for the FLEX Case Study. Three key CTRs, unmet in the current Air Force prototype, emerged in the analysis. These included requirements to

1.) adjust the viewpoint (level of detail),
2.) focus attention on the key decision variables, and
3.) compare response options in terms of potential consequences.

First, the TDO needed a way to "step back" from the detailed data with an overview of tanker operations. This was, in part, a response to the time horizon of the TDO's decisions and the varying degrees of timeliness and precision connected with the updates to the database. Small changes to the published ATO which must occur rapidly (e.g., last-minute re-routing of a mission to another tanker for refueling) are handled in the air by forward controllers. The TDO makes decisions involving a somewhat longer decision horizon and needs to work with an aggregated display of the entire ATO day. Second, the TDO needed a display simultaneously presenting all the critical decision factors. The working group participants complained that key information was distributed across several displays, requiring the user to jump around and make notes off-line. Third, the TDO needed a means of comparing the effects (e.g., changes in critical values) of choosing one option over other feasible options. Answering these requirements without sacrificing access to detail became the central goal of the CSE interface re-design.

Phase Three - Developing an Integrated System Requirements Document

The CTR is a statement expressing either the nature of the *input* required for a human decision-making task or the content of the *output* required from that task. To facilitate reviews and inspection, the CTRs must be integrated into the system requirements document. For example, given a functional requirement to monitor a situation, the statement of the related CTR (bold face) might take the generic form in Figure A-6. Not all CTRs may be represented in this discrete task format; these may be included in the accompanying diagrams and narrative descriptions that preface requirements specifications. Furthermore, the level of detail represented in requirements documents varies based on the format specified and the overall complexity of the system under consideration. Regardless of the granularity selected, CTRs should contribute to the correctness, clarity, completeness, verifiability, consistency, comprehensibility, modifiability, and traceability of the requirements document.

The FLEX Case Study - Integrating the CTRs in the FLEX System/Segment Specification (SSS) Document

The Department of Defense development standard for software systems, DOD-2167A, specifies the format and content of system-level requirements documented in a system/segment specification (SSS) document. Although the FLEX case study focused on the decision activities of the Tanker Duty Officer, the CTRs had to be identified and represented in the higher level format of the FLEX SSS. This integration involved distilling the findings from the requirements review presented in Appendix E and matching them to the relevant system specifications in the existing FLEX SSS. In many cases, the FLEX SSS already contained statements which incorporated the content of the CTR.

Purpose

This system capability provides the facilities which enable the user to review and monitor data which will facilitate the analysis of the impact of changes in available resources or external environment on the capability to accomplish Task *X*.

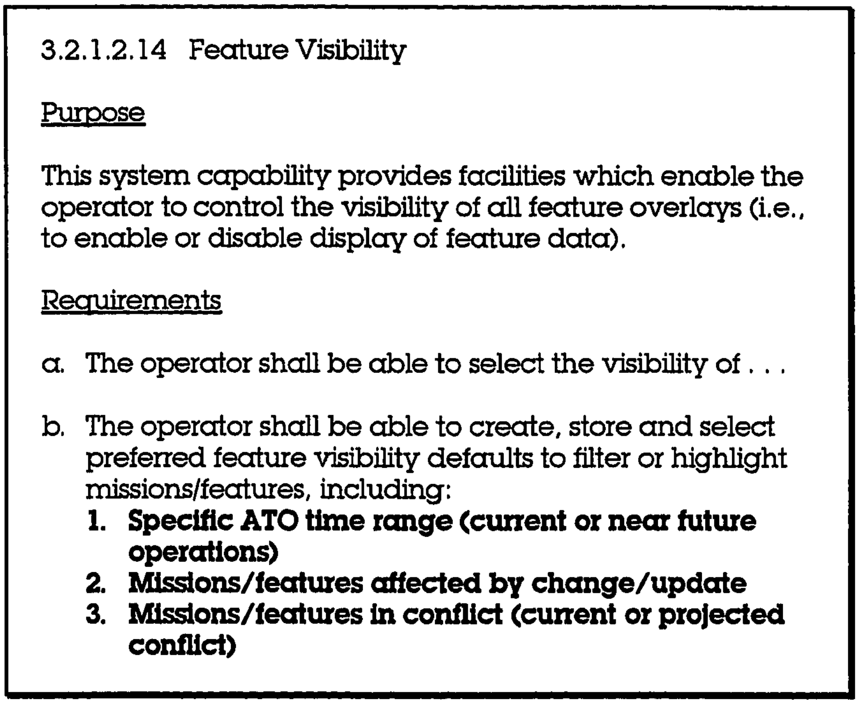
Requirements

a. The system shall provide facilities which permit the operator to detect changes in available operational resources which exceed a previously defined threshold.

- **The system shall provide facilities to display and compare the increase and/or decrease in Factor *F* during time *t*.**

Figure A-6: Example of a CTR Integrated in a System Requirements Document

Occasionally, the statements were modified to improve their precision. In addition, items were appended to stated requirements to detail functionality specified by identified CTRs (Figure A-7). Appendix F presents examples from the integrated System/ Segment Specification (SSS) for the FLEX Case Study.



3.2.1.2.14 Feature Visibility

Purpose

This system capability provides facilities which enable the operator to control the visibility of all feature overlays (i.e., to enable or disable display of feature data).

Requirements

a. The operator shall be able to select the visibility of . . .

b. The operator shall be able to create, store and select preferred feature visibility defaults to filter or highlight missions/features, including:
   1. **Specific ATO time range (current or near future operations)**
   2. **Missions/features affected by change/update**
   3. **Missions/features in conflict (current or projected conflict)**

Figure A-7: Example of a CTR Integrated in the FLEX System/Segment Specification Document

Phase Four - Translating Requirements into HCI Design Concepts

By the far, the most debatable and least prescriptible aspect of development is the process by which system requirements are translated into HCI designs. Viewed from a process perspective, the inputs represent the design goals derived not only from the formal requirements specifications, but also from the knowledge (e.g., domain, users, organizations, tasks, and situations, etc.) that guided the development of the formal specifications (Figure A-8). The translation process involves the interpretation of requirements (design goals) with respect to specific HCI design features. These features include task allocation (fixed and dynamic), information presentation (form, content, granularity, etc.), interaction characteristics (input/output, manipulation, etc.), and interface characteristics (windows, menu structures, etc.). Verification and validation of the design is accomplished by checking the features against requirements and vice versa. Ultimately, validity depends upon the extent to which the formal requirements specification represents the performance requirements of the operational domain and the informal knowledge.

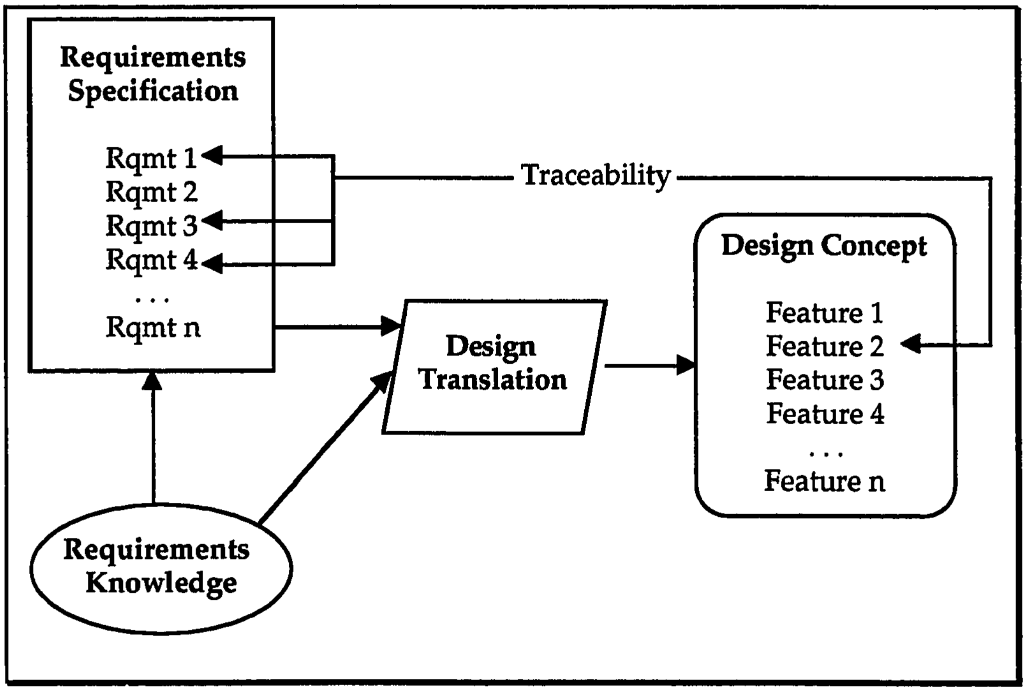


Figure A-8: Translating Requirements into an HCI Design Concept

There is a tendency to view the HCI-related components of the system as superficial and external to system functioning. This is due, in part, to developer's incorrectly interpreting the desirable engineering principles of modular design. While the user-machine interface (UMI) is rightly treated as discrete component in the systems architecture with physical and logical interfaces to the various functional modules, the interaction of the human user and the computer impacts all of the problem-related functions of the system. For example, even where computation tasks are allocated to the machine, the HCI design must consider such factors as

- How the computational factors will be input
  - manually? automatically?
- If manual input is required,
  - how will the user know what input is required?
  - how will the information be input by the user?
  - how will the user know if the input was correct?
- If the user must infer something based upon the computation,
  - how will the results be conveyed to the user?
  - how will the user explore the contents and impacts of the result?

Relegating HCI design decisions until after the functional design is complete limits the ability of the designer to meet requirements. Furthermore, while this approach might appear to optimize the machine performance, it provides no assurance of optimal *system performance* in the operational environment. The result can be an elegant algorithmic solution that cannot be used or is used incorrectly, degrades performance, and/or whose power is never realized in the operational context.

Traditionally, HCI design guidelines focused on the usability of the interface (i.e., searching, selecting, perceiving, etc.) rather than the usability and utility of the system with respect to larger task goals (i.e., making tactical or strategic decisions, assessing trends, etc.). In recent years, recognition of the interdependence of information presentation and interaction with the performance of problem-solving tasks motivated efforts to expand HCI design guidance to address these cognitive issues. Smith and Mosier (1986) developed the first comprehensive guidelines document. Although, most of the work focuses on UMI issues and does not incorporate recent technological advances, these guidelines remain a standard source. A key feature in Smith and Mosier's approach is the citation of HCI research to support the stated guidelines and indicate exceptions in practice. Much of the recent work in this area has sought to update and expand Smith and Mosier's guidelines rather than replace them.

The guidelines most applicable to human-computer cooperative decision-making in time-critical environments come primarily from research in two domains:

1.) **command & control** - Fernandes, 1992; Lewis & Fallesen, 1989; Obermayer & Fallesen, 1990
2.) **control systems** - Rasmussen & Pejtersen, 1993

Each of these documents addresses system support for functional cognitive tasks. Rasmussen and Pejtersen's guidelines are part of a cognitive engineering design methodology that incorporates models of the tasks and work domain (Rasmussen, Pejtersen & Goodstein, in press).

Two resources belong in unique categories. Gardiner and Christie (1989) present 162 design guidelines derived from the cognitive research in learning, mental models, memory, and related factors. While these guidelines focus more on usability design for UMI tasks, the organization of the guidelines within discussions of topics in cognitive psychology provides insights for applying the guidelines to decision tasks. From the perspective of goal-driven design, the checklists in Ravden and Johnson's (1989) usability evaluation method can provide the designer with design "targets." Despite their intended application in usability evaluation, Ravden and Johnson's checklists can be used to support a tighter coupling between requirements, design, and evaluation.

The various guidelines cited provide the designer with hints and insights to apply to the specific design goals identified in the previous phases. The designer filters and adapts these ideas to meet the hardware and software constraints of the system and to compliment the underlying analytical approach. Although much of the creative process of design may be viewed as an idiosyncratic art, the core of design is the *craft* of distilling design goals from requirements, using those goals to develop designs, and evaluating the designs based upon identified requirements.

The FLEX Case Study - Translating Requirements into an HCI Design Concept for the FLEX Tanker Module Prototype

The requirements identification process surfaced eight design goals (Appendix E, 6.0). Two of the goals involved requirements that were adequately addressed in the existing FLEX prototype and lay outside the specific interests of this research.[3] The remaining six belong to the general category of improving decision-making performance represented in the three FLEX CTRs listed in Phase 2. These six goals require designs that support

1. understanding operational & domain dependencies,
2. focusing on goal/decision-relevant information,
3. selecting the appropriate viewpoint (level of abstraction),
4. reducing mental workload,
5. improving situational awareness & understanding, and
6 comparing options.

In addition, the immediate benefit of improving performance, Goals 1 - 3 have the potential to enhance long-term performance by developing and reinforcing the mental models that produce a more robust decision-maker knowledge base.

The FLEX Tanker Case Study focused on the immediate benefits of performance improvement derived from the six design goals. Figure A-9 indicates the interdependencies associated with the individual goals. Research indicates that

---

[3] These two goals fall under the general headings of Decision Control & Guidance and Interface Operation & Error Control.

the quality of situation assessment and ability to preview the effects of decisions improves decision performance (Klein *et al*, 1992; Klinger *et al*, 1993; Raphael, 1991). In particular, the improving decision-maker's understanding of the causal dependencies that underlie a situation and the consequences of a given course of action can help to reduce decision error often associated with complex decisions (Cohen *et al*, 1985; Reason, 1990; Senders & Moray, 1991).

The keys to situational awareness and understanding lie in the decision-maker's ability to

1) filter the relevant situational cues from complex, rapidly changing data, and
2) combine the cues to make inferences about the situation (Andriole & Adelman, 1989).

Selecting the appropriate level of detail and focusing on decision-relevant information assists the filtering process; while an understanding of the operational and domain dependencies -- the causal networks -- provides a framework for combining information to make inferences. Relieving the decision-maker of certain detailed mental operations (e.g., calculations, table look-up operations, and various memory tasks) and providing mental organizers (e.g., decision-structured displays) permits the focus of mental resources on the critical decision tasks. Finally, the ability to compare options in terms of potential consequences of actions taken is enhanced by the decision-maker's focus and understanding.

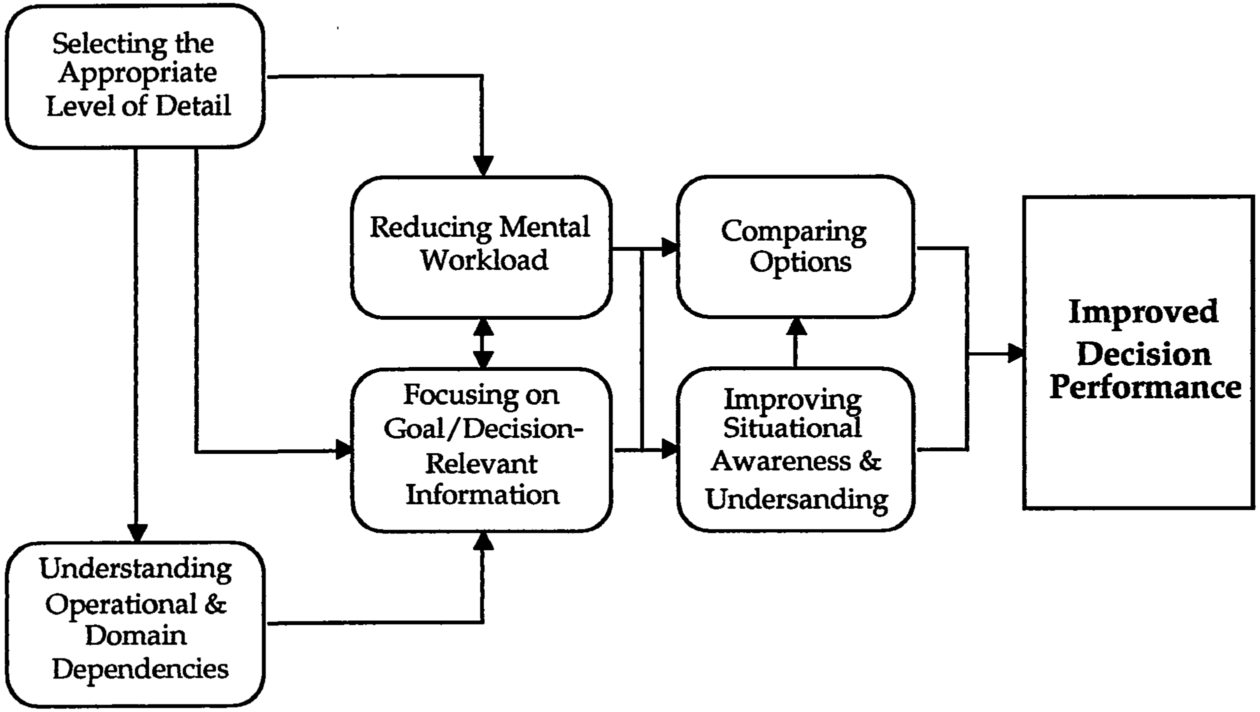


Figure A-9: Relationship of FLEX Design Goals to Overall Goal of Improved Decision Performance

The CTRs identified for the FLEX Tanker module in Phase 2 and incorporated into the six design goals above map to four CSE design principles. These principles, with the associated design goals in parenthesis, include:

- Presenting a system-level model relating the relevant decision variables to focus the decision-maker's attention and guide the selection of appropriate detail (Goals 1 - 6);
- Integrating all the key decision factors in one display to eliminate unnecessary jumping from screen to screen (Goals 2 - 5);
- Making the current system (i.e., tanker operations) state visible to highlight the areas requiring correction (Goals 2 - 5);
- Relieving the decision-maker of calculation and memory tasks (Goals 2, 4 & 6); and
- Making the consequences of options visible for comparison and evaluation (Goals 1, 2 & 6).

The first two principles were drawn primarily from the ecological interface design research by Jens Rasmussen and his colleagues (Rasmussen & Vicente, 1989; Vicente & Rasmussen, 1992) and represented in guideline form in Rasmussen & Pejtersen (1993) and Rasmussen *et al,* (in press). In addition, research on the design of integrative displays (Bennett *et al,* 1993) provided further insight into the ways decision cues can be combined in symbolic displays whose decision-aiding "emergent" features are only apparent in that combined form. Finally, the tactical decision-making research by MacMillan & Entin (1991) illustrated the decision performance value of unifying the key decision factors in a single window. The three remaining principles reflect guidance that may be found in all standard guideline sources.

The guidance from these principles drove the design of an additional window for the FLEX Tanker DO called ***Option View***. (Appendix G, Figure G-8).

The ***Option View*** window incorporates a number of HCI responses to the CSE principles identified. First, the window presents a high-level system model of current tanker operations displaying the active tanker missions at their orbit locations across the 24 hours of the ATO. The receiver contacts are mapped across time against the assigned tanker mission to highlight their flow in terms of density and timing. Conflicts are highlighted in red to draw attention; changes in the tanker or receiver missions are highlighted in yellow. The taskable fuel remaining is displayed above each tanker mission and relieves the decision-maker from having to make the calculation. Second, to facilitate comparison, two options may be compared simultaneously against the planned ATO. (The actual large-screen monitor used for the Air Force FLEX prototype would support comparison of more than two options.) The comparisons present the effects of allocations in terms of changes to the taskable fuel remaining, timing of receiver contacts, and density of assigned receivers against the tanker.

Phase Five - Developing an Interactive Prototype for Evaluation

The HCI design concepts developed through the CSE framework embody hypotheses about the effects of information presentation and interaction on human decision performance. The best means of evaluating these design hypotheses is to try them out by implementing a series of simple prototypes (usually paper-based prototypes evolving to an interactive computer-based prototype). At each stage, the proposed design can be reviewed against the current version of the requirements. Sponsors and operational users can respond to the prototyped design to refine the requirements base and assess the utility and usability of the proposed interface for the decision tasks.

Strategies for prototyping vary widely the depending upon the purposes of the prototype and the resources available to implement and evaluate the prototypes. Davis (1993) summarizes the purposes and development impacts of the two principles approaches to prototyping: the "throwaway" prototype and the "evolutionary" prototype. Where the throwaway prototype is often used for requirements exploration and then discarded (Boar, 1983), the evolutionary prototype is designed for iterative review and modification with the intent that it will ultimately result in a delivered system. Connell and Shafer (1989) present an approach to "evolutionary" prototyping that incorporates the principles of structured design and development necessary to ensure the prototype will evolve successfully into a quality software product. One over-riding principle guides prototyping development, regardless of the implementation strategy, ***never show the client features in prototype form which cannot be realized within the technological and budgetary constraints of the proposed development effort.***

The goal of a deliverable product can negatively impact the usefulness of an evolutionary prototype in the volatile early stages of problem definition and requirements identification. Despite the considerable advances in computer-aided software engineering (CASE) tools and programming libraries, some combination of throwaway and evolutionary prototyping is desirable for meeting development phase goals. The CSE framework elaborated here is adaptable to either "throwaway" or "evolutionary" prototyping depending upon the goals and resources of the developer.

The FLEX Case Study - Developing the FLEX Tanker Module Prototype

The FLEX ATTD is a technology demonstration program that is intended to evolve into a fielded system. Given the author's external role in the FLEX ATTD, the FLEX Tanker Case Study made use of a throwaway prototype to evaluate the HCI design impacts on decision performance. For evaluation and comparison, both the FLEX tanker module displays and the revised HCI design were implemented in an interactive prototype. The essential features of the existing FLEX windows were mocked-up to allow for rapid prototyping of the key decision factors presented in each window. The extensive searching, sorting and tailoring capabilities of these displays were not represented in order to focus the evaluation on the decision-making tasks rather than the interface manipulation tasks. The evaluation prototype was developed in SuperCard® on an Apple Macintosh IIci® with a high-resolution RGB color monitor. To facilitate non-intrusive, automated data collection, the software program includes routines to record time-stamped information about the user's interaction with the interface.

Phase Six - Evaluating the Prototype

Modeling the human-computer interaction aspects of a system supporting human decisionmakers in complex, dynamic, high risk environments presents an formidable set of challenges. Human performance in cognitive tasks is exceptionally resistant to representation in cleanly defined cause and effect models. The interaction strategies and technological features which comprise the HCI design generally cannot be linked directly to task performance -- let alone overall system performance. Moreover, once it is recognized that humans are not interchangeable components, it immediately becomes apparent that simple outcome models are insufficient. HCI evaluation in this context must include some representation and appraisal of the *processes* involved in task performance, as well as the outcome of that performance. Thus, HCI evaluation is almost always performed based upon a set of hypotheses that relate design features to changes effected upon processes which, in turn, effect changes in performance outcome.

In a requirements-driven design process, the judgments and decisions made during each phase determine the objectives of the analyses and evaluations required to support those decisions. These phase-related objectives further define the scope and boundary of the evaluation in terms of the extent to which a given study considers organizational interactions and environmental context as well as the level of detail with which these factors are represented. Ehrhart (1993) summarizes the phase-related objectives and HCI design evaluation products that apply at each phase in the system development life cycle (SDLC). Figure A-10 maps the inputs and high-level evaluation goals (bold lines) for each phase.[4]

As outlined above, the evaluation of the prototype should track to the design goals as defined by the requirements. Two principle evaluations should be conducted at each level of prototyping: 1) verification of design implementation of HCI requirements, and 2) validation of design implementation's effectiveness in terms of interface usability and utility. In the cost- and time-sensitive environment of systems development, computer-based interactive prototypes provide an opportunity for rapid, low-cost focused evaluation. In addition, several methods are available for examining interaction processes through automated capture and analysis of interaction protocols (Smith *et al*, 1993). This method affords direct observation of the human-computer decision performance with varying levels of internal and external validity.

Evaluation guidance for the practitioner can be found in sources ranging from the traditional human factors research (Bailey, 1989) to those focusing specifically on decision support system (Adelman, 1992). For complex systems, David Meister's (1985) comprehensive survey of behavioral analysis and measurement methods brings together human factors and cognitive psychology while maintaining a systems engineering perspective. In the usability category, Ravden and Johnson's (1989) usability method lives up to its claims to be practical by using checklists that may be derived directly from requirements documents. Finally, Nielsen's (1993) usability approach presents a variety of

---

[4] While it is also possible to model the problem definition --> requirements specification input and evaluation loop, the specified requirements provide the standard for evaluation.

evaluation methods and links usability to cost savings in the workplace. These sources indicate that a well-designed small study can provide the appropriate information for early design stages and assist in developing the measures required for more comprehensive evaluation in later phases of development.

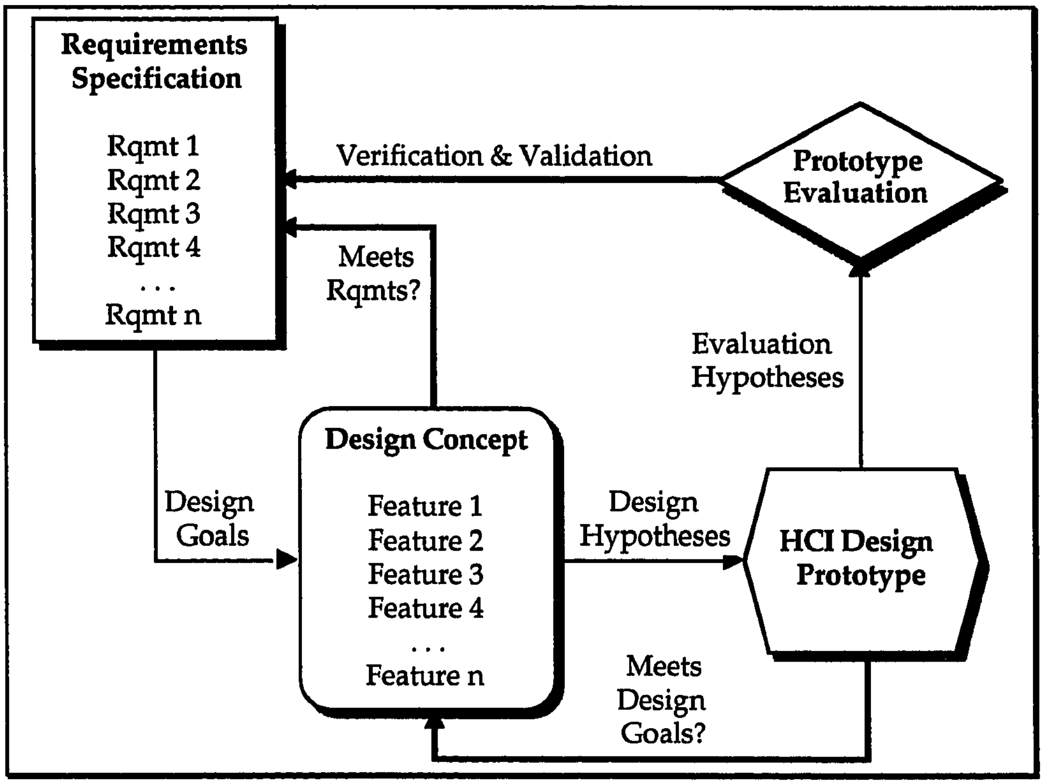


Figure A-10: Relationship of Process Inputs to Evaluation Goals at Each Development Phase

The FLEX Case Study - Evaluating the FLEX Tanker Module Prototype

The fundamental hypothesis of the CSE design framework is that using the methods should highlight the critical cognitive task requirements and, by guiding the translation of these requirements into design concepts, result in changes in the system which, in turn, result in changes in task performance. The evaluation of the FLEX Tanker Module Prototype sought to validate the CSE framework by demonstrating an improvement in decision performance along three dimensions: situational awareness & understanding, option evaluation, and cognitive workload. These dimensions incorporated the six design goals identified in Phase 4. Figure A-11 maps the specific hypotheses and related measures to these three dimensions. A brief discussion of the evaluation approach selected follows; further details on the evaluation and results may be found in Ehrhart (1994).

Hypotheses & Measures

As indicated previously, the plan for evaluating the FLEX HCI design was built upon a multi-dimensional view of the factors contributing to effective decision-making performance. The fundamental hypothesis for evaluation may be stated as follows:

> HCI designs based upon the CSE framework for identifying and specifying cognitive task requirements will result in improved decision-making performance.

This fundamental hypothesis was broken down into measurable factors with respect to three dimensions: situational awareness and understanding, option evaluation and selection, and cognitive workload. Each dimension was represented by one or more design goals that, in turn, were the subject of one or more sub-hypotheses and measures. The dimensions are discussed in turn below.

Dimension 1: *Situational Awareness & Understanding*

- Design Goal: Design presentation of information to highlight and relate key decision factors at the appropriate level of abstraction to relieve decision-makers from the requirement to accomplish this integration in their heads.

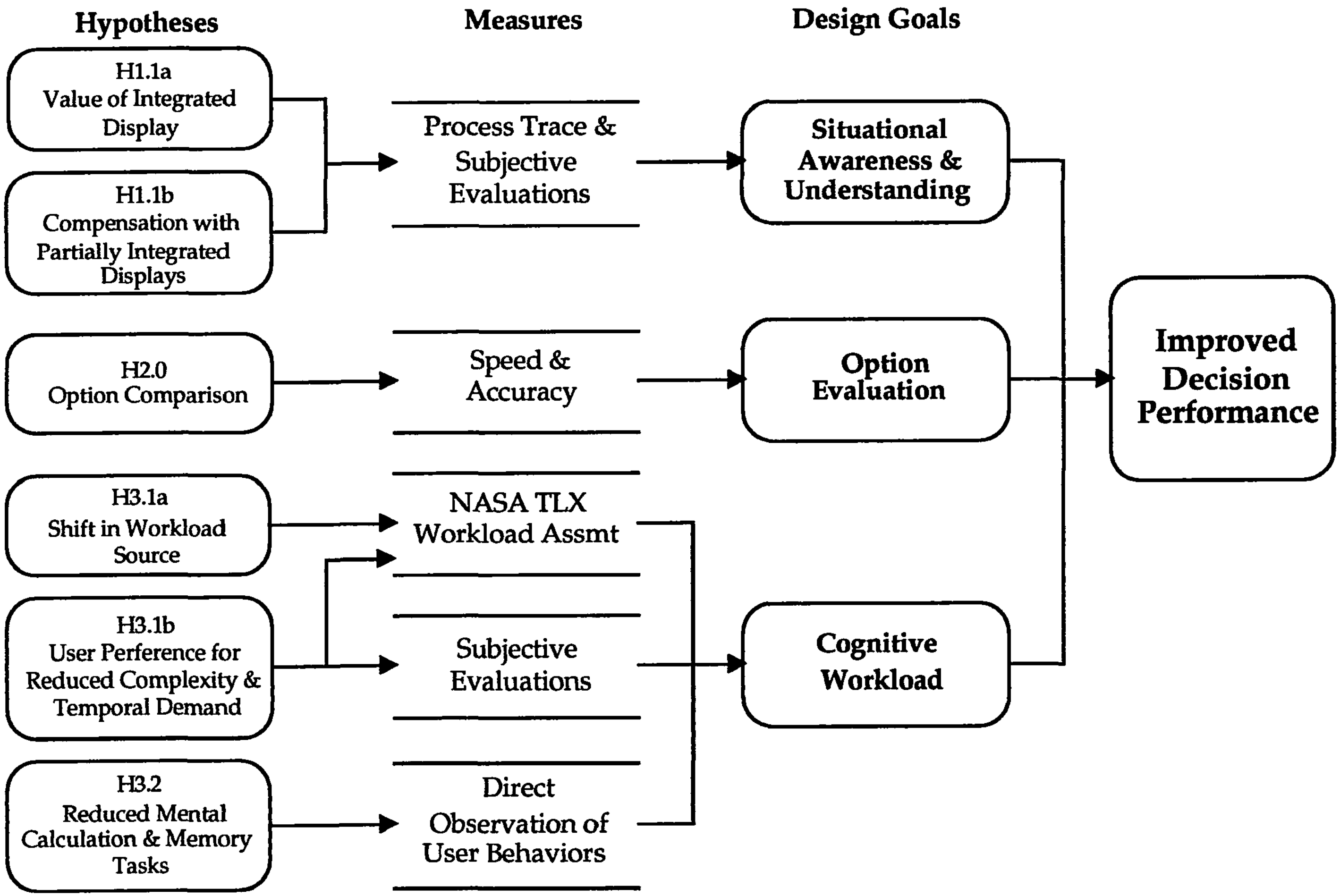


Figure A-11: Relationship of FLEX Evaluation Hypotheses & Measures to the HCI Design Goals

Hypothesis 1.1a: Decision-makers presented an integrated model of the "system" and critical decision variables will more accurately focus their information search than those not supplied with the integrated model display.

Hypothesis 1.1b: In the absence of a fully integrated model display, decision-makers will compensate by selecting the displays which partially integrate key variables.

Measures:

1. Time-stamped Process Trace of Information Views Used
   (comparison with decision model of where critical decision information is located)
   - comparison of mean frequency of window selection
   - process trace (precisely where user went when)
   - comparison of mean duration (in seconds) spent viewing each window
2. Subjective Interface Evaluations
   (comparison of interface/task means presenting users rating of specific window's usefulness in four decision tasks)
   - Problem Identification
   - Situation Assessment
   - Option Evaluation
   - Option Selection

Dimension 2: Option Evaluation

- Design Goal: Design information presentation and interaction to allow comparison of two or more options in terms of their consequences across time.

  Hypothesis 2.0: Displaying the changes in the critical variables to allow simultaneous comparison of two or more options will improve option evaluation and selection performance.

Measures:

1. Speed
    - (comparison of mean trial and sum times to make individual decision by interface)
2. Accuracy
    - comparison of mean score on selection of "better" option across trials, users, and interfaces
    - comparison of ANOVA on scores across trials, users, and interfaces ("better" option determined by previously established experts' model rating options based on taskable fuel remaining & receiver "density" function)

NOTE: Interface exposure order effects were compared to evaluate potential task/interface learning interaction across sessions.

Dimension 3: Cognitive Workload

- Design Goal: Reduce decision-maker's experience of mental demand and time-pressure by designing the information presentation as a "system model" representing and relating critical decision variables.

Hypothesis 3.1a: When other task factors are held constant, the perceived workload associated with time-pressure and problem complexity will be greater for decision-makers working without integrated displays.

Measure:

NASA-TLX workload assessment.[5]

- comparison of the percentage of total workload attributed to temporal and mental demand depending upon interface used

---

[5] NASA TLX is a subjective rating of the user's perception of the source of task workload across multiple dimensions (e.g., mental demand, temporal demand, own performance, frustration, effort, etc.). The user rates each dimension on a set of common scales after each trial. Dimensions are weighted after trials are completed using paired comparisons.

Hypothesis 3.1b: The subjective evaluation of interfaces will favor those interfaces associated with lower cognitive workload ratings (i.e., those that reduce task complexity in terms of mental and temporal demand).

Measures:

1. NASA-TLX workload assessment
   - mean percentages (described above) by interface
   - mean total workload by interface
2. Subjective Interface Evaluations
   - comparison of mean subjective evaluations interface effectiveness across decision tasks (problem identification, situation assessment, option evaluation, option selection)
   - review of open-ended written & verbal impressions of interfaces (audio recording of discussion after final session) vis-a-vis task requirements

- Design Goal: Display changes in the critical variables to relieve the decision-maker of the extra cognitive workload involved in mentally simulating the comparative effects of the options. Tasks, such as calculation of numerical values (e.g., fuel remaining), should be allocated to the computer to relieve users of mental calculation.

Hypothesis 3.2: Decision-makers provided integrated displays (i.e., those presenting calculations of all key variables) for comparing the options will not make off-line notes to support their mental simulations.

Measure:

Direct observation

- collection of session materials for review to determine if users had to take notes and calculate values while using the interface

Subjects

To increase the validity of the evaluation with respect to the target domain, evaluation subjects were drawn from tanker squadron officers. All subjects (junior aircraft commanders and senior co-pilots) had an equivalent level of experience and training in tanker operations. The volunteer subjects were randomly assigned to one of four groups by the squadron operations center based upon scheduled availability.

Procedure

The evaluation involved a within-subjects, repeated measures design. The four subject groups were handled as two blocks, each interacting with both interfaces. Subjects participated in two interactive sessions, each comprised of 12 successive decision trials. The order of interface exposure was varied to control for interface learning effects (Table A-1). At the first session, the group completed background information forms and signed advised consent forms. Following this, subjects received instruction on the task domain (force-level re-planning decisions), fundamental processes of decision-making, the NASA TLX workload assessment forms, and the interface for that session. Before the experimental trials began, subjects were allowed to practice for 15 minutes on an example trial using the interface.

INTERFACE	INTERFACE EXPOSURE ORDER	
	Block 1	**Block 2**
Interface A (original)	1st Session	2nd Session
Interface B (CSE framework)	2nd Session	1st Session

Table A-1: FLEX Case Study Data Collection Design

The total experimental interaction was time-constrained, but individual trials could be completed at the user's pace. The following data was collected on each trial:

- Time - trial start, trial stop, decision selection (automated);
- Process - time stamped trace of screens viewed (automated);
- Choice - option chosen (automated); and
- Workload - NASA TLX trial form (manual).

After completing all 12 trials, the users completed forms to provide additional subjective assessment of the interface/interaction:

- Workload - NASA TLX interface form;
- Task Usefulness - screen evaluations; and
- Usability/Utility - free-form interface evaluations.

The second interface session was conducted on the following day. The session began with an introduction to the second interface and a 15-minute practice session. The experimental interaction was conducted to match the first session. At the close of the second session there was an open discussion to allow the subjects to make comments about the two interfaces, the decision tasks, and the experimental process. Subjects were encouraged to describe their decision processes. This discussion was audio taped to permit later analysis for common themes.

The twelve trials were developed using input from several Air Force officers with recent experience in tanker operations. The data to support the trials (e.g., fuel requirements based on aircraft type, mission and route) was generated by the Automated Planning System (APS), a standalone operational prototype supporting the Combat Plans Division of the AOC in mission planning and ATO generation. In operational deployment, FLEX will receive the planned ATO for monitoring and execution from APS. The "best" option for each was identified in coordination with this team of officers as an expert judgment.

Results

Evaluation of CSE-based HCI design comprised investigations of decision-making performance and processes coupled with process and performance assessments for the decision support aspects of the HCI design. The analysis was based on objective and subjective measures collected in a controlled experiment involving domain-knowledgeable users. As presented in Ehrhart (1994), the CSE-based HCI design consistently produced more desirable decision-making performance. Subjects using the CSE interface arrived at decisions approximately 26% faster with a 12% improvement in decision accuracy. Analysis of the interaction between the interface used and the exposure order revealed significant changes in performance which may be due to support in the CSE interface for creating a returnable mental model of the complex interdependencies in the operational environment. The CSE interface users demonstrated more focused use of the interface for information review as reflected in 29% fewer window changes. Further review of the window usage revealed several trends regarding the use of graphical overview displays versus detailed data displays. The interaction effects noted in the objective performance measures were also significant for the objective process measures.

The decision support provided by the CSE interface resulted in a 20% reduction of task workload as measured by the NASA Task Load Index. The TLX analysis further revealed a shift of the source of workload from external stressors (mental and temporal demand) to the internal motivation factors measured as the user's own performance standards. Subjective evaluations of the individual interface windows provided additional support for the objective process findings regarding graphic overview and detailed data displays. Users uniformly rated the graphic overview windows higher than detailed data displays across the four decision tasks (problem identification, situation assessment, option evaluation, and option selection). When available, the ***Option View*** window received the highest scores; when ***Option View*** was unavailable, users rated the ***Marquee*** highest. This scoring shift matched the window selection shift noted in the objective process measures under similar circumstances.

The results of the experimental evaluation favored the CSE HCI design for all the objective and subjective measures. Overall, the evaluation demonstrated the benefits of using the CSE framework in three key areas:

- *Decision-Making Performance*
  - » Reduces decision performance time
  - » Improves decision accuracy
  - » Supports more focused, effective use of the interface
- *Decision Support Performance*
  - » Reduces workload overall
  - » Shifts source of workload to more positive internal performance standards
- *User Acceptance*
  - » Focus on cognitive task requirements results in a better match with the decision-maker's information presentation preferences.

Decision-makers using the CSE interface completed tasks faster with greater accuracy and used the interface to review information more effectively than when using the original interface. The difference in the source of workload from time stress to the more positive pressures of self-imposed performance standards also favored the CSE interface. Integrating the key decision information in the ***Option View*** window allowed the decision-makers access to the required information in the preferred graphical overview display format rather than wading through the detailed data. Finally, the users uniformly preferred the CSE interface for all the decision tasks involved in tanker replanning.

Cognitive Systems Engineering in the Development Life Cycle

As presented in the introduction and Figure A-1, the six phases comprising the CSE method are designed to fold into the typical development life cycle. The methods presented in this handbook focus on using available resources to achieve the maximum design benefits with the least impact to development costs and schedule. At each stage in systems design and development, the various decision-makers involved require information inputs from both the analyses of requirements (system objectives, functions, tasks, operational capabilities) and evaluations of performance and effectiveness characteristics (current and potential). As discussed in the previous section and presented in Figure A-12, feedback from evaluation provides the designers and project management the necessary information to ensure the system delivers the desired level of functionality and meet cost and schedule requirements. Evaluation feedback also supplies the verification and validiation critical to assuring end-user satisfaction and sponsor acceptance.

Feedback is a course correction device. Early evaluation allows design modification during the initial life cycle phases when the cost to modify is lower. For the design team, evaluation is also a discovery process. Findings from the evaluation provide input for requirements and design modification and help to set MOP and MOE benchmark targets for later system-level evaluations. Evaluation feedback informs not only the design of the particular functions and features considered, but also provides input for the design of related components. For the project manager, evaluation feedback is a critical part of project planning and control. Early evaluation flags potential problems which may require cost, schedule or, in some cases, contract modification.

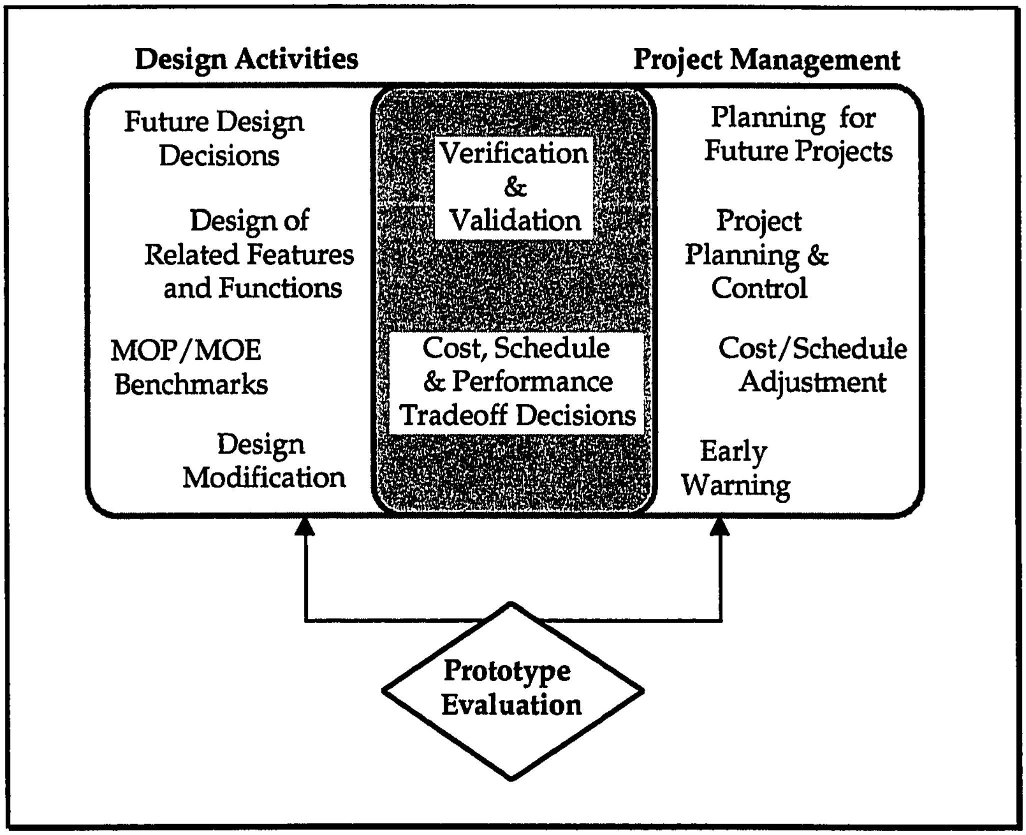


Figure A-12: Benefits of Evaluation Feedback to Development

Ehrhart (1994) presents initial evaluation of the impacts of CSE design on both the development process and the end product. For both the designer and manager, incorporating CSE activities into the development process assures a better match to the operational need by capturing a more robust set of functional and non-functional requirements. This understanding supports informed decision-making when design tradeoffs must be made during development life cycle.

The CSE Design Framework synthesizes the findings from numerous studies in human-computer interaction, decision-making, and multimodal communication to permit the integration of this body of knowledge into the design of systems to support human-computer cooperative decision-making. This handbook, as an initial prototype, attempts to present CSE design in a usable format for the system design practitioner. It is intended as a living document, with future iterations reflecting both empirical research and practical experience. The author actively encourages feedback on any and all aspects of the method and handbook.

Handbook References

Adelman, L. (1992). *Evaluating Decision Support and Expert Systems.* New York: John Wiley.

Andriole, S. J. (1990). *Information System Design Principles for the 90s: Getting it Right!* Fairfax, VA: AFCEA International Press.

Andriole, S. J. and L. Adelman. (1989). *Cognitive Engineering of Advanced Information Technology for Air Force Systems Design & Development: Prototypes for Air Defense Systems Design & Development.* [Technical Report prepared for Air Force Systems Command, Armstrong Aerospace Medical Research Laboratory (AAMRL), Human Engineering Division (HED), Wright-Patterson AFB, OH 45433]. Marshall, VA: International Information Systems, October.

Andriole, S.J. and L.S. Ehrhart. (1990). *Intelligent Decision Support for Electronic Warfare: The IDSSEW Prototype.* Fairfax, VA: Center for Excellence in Command, Control, Communications & Intelligence, School of Information Technology & Engineering, George Mason University, September.

________. (1993). "Adaptive Decision-Aiding Via Intelligent Interfaces." *Proceedings of the 10th Annual Conference on Command and Control Decision Aids.* (National Defense University, Fort McNair, Washington, D.C., June 30 - July 1, 1993.) Fairfax, VA: SAIC.

Bailey, R. W. (1989). *Human Performance Engineering: Using Human Factors/Ergonomics to Achieve Computer Usability.* Englewood Cliffs, NJ: Prentice-Hall.

Bennett, K. B., M. L. Toms, and D. D. Woods. (1993). "Emergent Features and Graphical Elements: Designing More Effective Configural Displays." *Human Factors*, 35(1), 71-97.

Boar, B. (1983). *Application Prototyping: A Requirements Definition Strategy for the 80s.* New York: John Wiley.

Boff, K. R. and J. E. Lincoln. (1988). *Engineering Data Compendium: Human Perception and Performance.* Armstrong Aerospace Medical Research Laboratory (AAMRL), Wright-Patterson AFB, OH.

Cohen, M. S., B. B. Thompson, and J. O. Chinnis, Jr. (1985). *Design Principles for Personalized Decision Aiding: an Application to Tactical Air Force Route Planning.* RADC-TR-85-140. Final Technical Report. McLean, VA: Decision Science Consortium, Inc., July.

Connell, J. L. and L. B. Shafer. (1989). *Structured Rapid Prototyping: and Evolutionary Approach to Software Development.* Englewood Cliffs, NJ: Prentice-Hall.

Cuomo, D. L. and C. D. Bowen. (1993). *Measures of User-System Interface Effectiveness: An Encoding Scheme and Indicators for Assessing the Usability of Graphical, Direct-Manipulation Style User Interfaces.* [MTR 92B0000047V3] Bedford, MA: MITRE Corporation, January.

Dept. of Defense. (1987). *Military Handbook. Human Engineering Procedures Guide.* DOD-HDBK-763. Washington, DC: Dept. of Defense, February.

Ehrhart, L. S. (1994). *Cognitive Systems Engineering: Developing Human-Computer Interaction Designs for Decision Support.* Ph.D. Dissertation. Fairfax, VA: School of Information Technology & Engineering, George Mason University.

_________. (1993). *Tasks, Models & Methods: System-Level Evaluation of Human-Computer Interaction (HCI) for Command & Control Decision Support.* Technical Report #MISE-3-1-3. [Prepared for Rome Laboratory/C^3AA under Contract #F30602-92-C-0119]. Fairfax, VA: C^3I Center, George Mason University, March.

_________. (1990). "New Approaches to Human-Computer Interaction Research and Design for Decision Aiding Systems." In *Proc. IEEE International Intelligent Control Symposium,* Philadelphia, PA, 5-7 September. New York: IEEE.

Fernandes, K. (1992). *User Interface Specifications for Navy Command and Control Systems, Version 1.1.* Naval Command, Control, and Ocean Surveillance Center; Research, Development, Test and Evaluation Division, June 1992.

Fleishman, E. A. and M. K. Quaintance. (1984). *Taxonomies of Human Performance: the Description of Human Tasks.* New York: Academic Press.

French, W. L. and C. H. Bell, Jr. (1973). *Organization Development: Behavioral Science Interventions for Organization Improvement.* Englewood Cliffs, NJ: Prentice-Hall.

Gardiner, M. M. and B. Christie, (Eds.). (1989). *Applying Cognitive Psychology to User-Interface Design.* New York: John Wiley & Sons.

Gowin, D. B. and J. D. Novak. (1984). *Learning How to Learn.* New York: Cambridge University Press.

Hammond, K. R. (1990). "Intuitive and Analytical Cognition: Information Models." In A. P. Sage (Ed.), *Concise Encyclopedia of Information Processing in Systems and Organizations.* Oxford, UK: Pergamon Press.

Heimrich, R. L. (1988). *Testimony before the U.S. House of Representatives, Committee on Armed Services on the Subject of the USS VICENNES Downing of Iranian Air Flight 655.* Congressional Record. October 6.

Hermann, C. F. (1972). "Threat, Time, and Surprise: A Simulation of International Crisis." In Hermann, Charles F., (Ed.), *International Crises: Insights from Behavioral Research.* New York: Free Press

Janis, I. J. (1989). *Crucial Decisions: Leadership in Policy Making and Crisis Management.* New York: Free Press.

Klein, G. A., J. Orasanu, R. Calderwood, and C. E. Zsambok, (Eds.). (1992). *Decision Making in Action: Models and Methods.* New York: Ablex.

Klinger, D. W., S. J. Andriole, L. G. Militello, L. Adelman, and G. A. Klein. (1993). *Designing for Performance: a Cognitive Systems Engineering Approach to Modifying an AWACS Human-Computer Interface. Final Technical Report* [Contract # F33615-90-C-0533; Department of the Air Force, Armstrong Laboratory, Wright-Patterson AFB, OH 45433]. Fairborn, OH: Klein Associates, Inc., March.

Lewis, H. V. and J. J. Fallesen. (1989). *Human Factors Guidelines for Command and Control Systems: Battlefield and Decision Graphics Guidelines.* [ARI Research Product 89-01] Alexandria, VA: Army Research Institute for the Behavioral and Social Sciences, March.

MacMillan, J. and E. B. Entin. (1991). "Decision-Oriented Display Design and Evaluation." In *Proceedings of the 1991Symposium on Command and Control Research*, June 24-25, 1991, National Defense University, Ft. Lesley J. McNair, Washington, DC. McLean, VA: SAIC, December.

McFarren, M. R. (1987). *Using Concept Mapping to Define Problems and Identify Key Kernels During the Development of a Decision Support System.* Master Thesis. Wright Patterson AFB, OH: Air Force Institute of Technology, School of Engineering.

Meister, D. (1985). *Behavioral Analysis and Measurement Methods.* New York: John Wiley, 1985.

_________. (1991). *Psychology of System Design.* Amsterdam: Elsevier.

Metersky, M. L. (1993). "A Decision-Oriented Approach to System Design and Development." *IEEE Transactions on Systems, Man and Cybernetics*, July/August 23(4),1024-1037.

Montanaro, G. D., W. J. Schroeder, B. Yamrom, W. Lorensen, D. Moitra, and P. M. Meenan. (1991). *Graphical Requirements for Force Level Planning, Volume 1. Final Technical Report.* [RL-TR-91-239, Vol I. Contract # F30602-89-C-0078. Prepared for Rome Laboratory, Advanced Concepts Branch, Griffiss AFB, NY.] Schenectady, NY: General Electric Company, September.

Moitra, D., G. D. Montanaro, C. Chalek. (1991). *Graphical Requirements for Force Level Planning, Volume 2. Final Technical Report.* [RL-TR-91-239, Vol II. Contract # F30602-89-C-0078. Prepared for Rome Laboratory, Advanced

Concepts Branch, Griffiss AFB, NY.] Schenectady, NY: General Electric Company, September.

Nielsen, J. (1993). *Usability Engineering*. Boston: Academic Press.

Norman, D. A. (1988). *The Psychology of Everyday Things*. New York: Basic Books.

Obermayer, R. W. and J. J. Fallesen. (1990). *Human-Computer Interaction in Tactical Operations: Design for Effective Human-Computer Dialogue.* [ARI Research Product 90-31] Alexandria, VA: Army Research Institute (ARI) for the Behavioral and Social Sciences, September.

Perlman, G. (1989). *User Interface Development.* [SEI Curriculum Module SEI-CM-17-1.1. Prepared for SEI Joint Program Office, ESD/AVS, Hanscom AFB, MA.] Pittsburgh, PA: Carnegie Mellon University, November.

Pew, R. W. (1988). *Testimony before the U.S. House of Representatives, Committee on Armed Services on the Subject of the USS VICENNES Downing of Iranian Air Flight 655.* Congressional Record. October 6.

Raphael, T. D. (1991). *Cognitive Flexibility and Innovation in the Problem Definition Process for Tactical Command and Control.* [D-39W-91 (Revised). Prepared for Logistics Research Group, Air Force Human Relations Laboratory, Wright-Patterson AFB, OH 45433-6573.] Falls Church, VA: Mystech Associates, May.

Rasmussen, J. (1986). *On Information Processing and Human-Machine Interaction: an Approach to Cognitive Engineering.* New York: North-Holland.

Rasmussen, J. and A. M. Pejtersen. (1993). *Mohawc Taxonomy Implications for Design and Evaluation.* [Risø-R-673(EN)] Roskilde, Denmark: Risø National Laboratory, January.

Rasmussen, J., A. M. Pejtersen, and L. P. Goodstein. (in press). *Cognitive Systems Engineering.*.

Rasmussen, J. and K. J. Vicente. (1989). "Coping with Human Errors through System Design: Implications for Ecological Interface Design." *International Journal of Man-Machine Studies*, 31, 517-534.

Ravden, S. J. and G. I. Johnson. (1989). *Evaluating Usability of Human-Computer Interfaces: a Practical Method.* New York: John Wiley, 1989.

Reason, J. (1990). *Human Error*. Cambridge, UK: Cambridge University Press.

Senders, J. W. and N. P. Moray. (1991). *Human Error: Cause, Prediction, and Reduction.* Hillsdale, NJ: Lawrence Erlbaum.

Smith, J. B., D. K. Smith, and E. Kupstas. (1993). "Automated Protocol Analysis." *Human-Computer Interaction*, 8, 101-145.

Smith, S. L. and J. N. Mosier. (1986). *Guidelines for Designing User Interface Software.* ESD-TR-86-278. Bedford, MA: MITRE Corporation.

Vicente, K. J. and J. Rasmussen. (1992). "Ecological Interface Design: Theoretical Foundations." *IEEE Transactions on Systems, Man, and Cybernetics*, 22(3).

Wohl, J. C. (1981). "Force Management Decision Requirements for Air Force Tactical Command and Control." *IEEE Transactions on Systems, Man, and Cybernetics*, 11(9).

Appendix B:

Requirements Identification Guidance Tables

Table B-1: Situational/Environmental Context Categories
(Meister, 1991; Rasmussen, 1986)

Determinate	Moderately Stochastic	Severely Stochastic	Indeterminate
Characteristics	Characteristics	Characteristics	Characteristics
• Given situation (initial condition) has only one significantly probable outcome.	• Given situation (initial condition) has only limited number of qualitatively similar outcomes with a significant probability of occurrence.	• Given situation (initial condition) has large number of qualitatively similar outcomes with a significant probability of occurrence.	• Given situation (initial condition) does not provide much information about possible outcomes. No outcome can be identified as significantly more probable.
Examples	Examples	Examples	Examples
• Common mechanical systems • Highly institutionalized social systems • Control systems	• Genetic processes • Variability in mechanical systems due to variable dimensions of machine parts • Batter's performance in baseball game	• Conflicts between humans • Weather • Political elections	• Psychotic human behavior • Political alliances • Fashion fads
Response Strategies	Response Strategies	Response Strategies	Response Strategies
• Skill- Based & some Rule-Based • System controls performance is by manipulating initial conditions • Efforts focus on optimizing outcome	• Rule-Based & some Skill-Based • Highly constrained outcomes induce attempts to manipulate initial conditions • Similar to deterministic, efforts focus on optimizing the *expected value* of outcome	• Rule-Based & Knowledge-Based • Outcome not controlled precisely by manipulating initial conditions • Detailed planning less useful due to unpredictability of evolving situation • Efforts focus on preparing for unfavorable outcomes & maintaining an ability to rapidly exploit opportunities	• Knowledge-Based • Requires intuitive approach to protect against disaster • Efforts focus on learning enough about the system (situational/environmental context) to classify it as another type

Table B-2: System Design Issues by Contextual Category

(Meister, 1991; Rasmussen, 1986; Vicente & Rasmussen, 1992)

Determinate	Moderately Stochastic	Severely Stochastic	Indeterminate
Goals	Goals	Goals	Goals
• Automate responses to a limited, highly structured set of triggering events • Rapid & effective response to relatively unchanging environment	• Develop a set of general procedures (without implementation details) to highly constrained set of triggering events • Maintain operational consistency & control under routine, but critical, performance demands	• Develop means to make most efficient use of resources in a succession of varying short-term situations; rapidly & effectively exploit opportunities • Overall efficiency gained by trading off internal consistency & automation for versatile, adaptive response	• Develop means to ensure effective response to potentially novel situations; support creativity • Support maximum flexibility and adaptiveness
Potential Errors	Potential Errors	Potential Errors	Potential Errors
• Rigid response to major changes and/or novel events	• Maintaining consistency may result in application of inadequate procedures when environment changes	• Potential misallocation of resources due to latency between recognition of situation & internal readjustment	• Flexibility may be accomplished at the cost of control • Focus on innovation may reduce ability to make use of experiential learning
Information Requirements	Information Requirements	Information Requirements	Information Requirements
• Detailed data highly important • Information about structure is less important • Input & output values specify system - initial conditions - past results	• Detailed data highly important • Information about structure is more valuable than in deterministic • Structural knowledge used to develop predictive models - possible outcomes - transition probabilities	• Detailed data less important; need symbolic displays to relate functional relationships • Structure may be inferred with sufficient information, but small amounts of data can be misleading; overview displays • Multiple levels of abstraction available to assist adaptive cognitive control requirements	• Structure is all important; overview displays provide externalized mental models • Detailed data & facts useful for empirical learning • Multiple levels of abstraction available to assist adaptive cognitive control requirements

Table B-3: Degree of Structure and Boundedness in Decision Context & Tasks

(Fleishman & Quaintance, 1984; Meister, 1991; Rasmussen *et al*, in press)

Closed Structured	Semi-Structured	Open Unstructured
Boundedness • Tractability Minimal demands, easily mastered • Representativeness & Reliability Accurately represents reality	**Boundedness** • Tractability Manageable by highly trained & motivated expert • Representativeness & Reliability Generally representative & reliable	**Boundedness** • Tractability Exceeds human ability to absorb and manipulate • Representativeness & Reliability Not well-understood and/or unreliable
Structure • Quantifiability Naturally or easily quantifiable • Availability All critical information readily availability	**Structure** • Quantifiability May be quantified without losing critical information or making difficult assumptions • Availability Some critical information unavailable ("known unknowns")	**Structure** • Quantifiability Not legitimately quantifiable • Availability Some critical information unavailable without user awareness ("unknown unknowns")

Table B-4: Level of Complexity in Decision Context & Tasks

(Fleishman & Quaintance, 1984; Meister, 1991; Rasmussen *et al*, in press)

Simple	Moderately Complex	Complex
Structural Features • Vertical Complexity None or very few hierarchical levels • Horizontal Complexity Few units or subsystems per level	**Structural Features** • Vertical Complexity Several hierarchical levels • Horizontal Complexity Several units or subsystems per level	**Structural Features** • Vertical Complexity Many hierarchical levels • Horizontal Complexity Many units or subsystems per level
Interdependency • No Dependency Functioning of unit/subsystem is unaffected by performance or non performance of other unit(s)/subsystem(s)	**Interdependency** • Moderate Dependency Functions can be performed, but may be enhanced or degraded by the performance or non-performance of other unit(s)/subsystem(s)	**Interdependency** • High Dependency Functions cannot be performed if other unit(s)/subsystem(s) perform poorly or not at all.

Table B-5: Developing Organizational/Doctrinal Profiles

(French & Bell, 1973)

Focus	Description & Examples	Information Sought	Investigation Methods
Total Organization (common mission and power structure)	• Investigation of organization as an entity. May also include relevant external (environmental) organizations, groups or forces, such as governmental agencies, lateral associations • Examples: manufacturing firm, hospital, school system, etc.	• What is the organizational culture? • What is the organizational climate -- open vs. closed, authoritarian vs. democratic, repressive vs. developmental, trusting vs. suspicious, cooperative vs. competitive? • How well do key organizational processes, such as decision making and goal setting function? • What kind and how effective are the organization's "sensing mechanisms" to monitor internal and external demands? • Are organizational goals understood and accepted?	• Questionnaire surveys • Interviews (group & individual) • Panel of representative members • Examination of organizational documents (policies, standards, etc.)
Large Subsystems of Large Organizations (complex & heterogeneous)	• Investigation of various organizational "slices": hierarchical level, function, location, etc. • Examples: Middle management group from diverse functional groups; functional groups with distributed management; everyone at one location, etc.	• All of the above, plus: • How does this subsystem view the whole and vice versa? • How does the subsystem fit into the organization? Are the subsystem goals compatible with organizational goals? • How do members relate to each other? • What are the unique demands? Are the organizational structures/processes related to the unique demands? Why? • What are major problems for subsystem and its subunits? What factors interfere with effective subsystem performance?	• Questionnaire and survey techniques recommended for large or dispersed subsystems • Interview and observation for supporting information

Table B-5: Developing Organizational/Doctrinal Profiles (cont.)

(French & Bell, 1973)

Focus	Description & Examples	Information Sought	Investigation Methods
Small Subsystems of Large Organizations (simple & relatively homogeneous)	• Typically formal work groups or teams with frequent, direct contact. May be permanent or ad hoc. • Examples: Top management team, any manager and key subordinates, committees, task force teams, etc.	• Questions on culture, climate, and attitudes are relevant, plus: • What are major problems of this group or team? • How can team effectiveness be improved? • How well do the member/leader relationships work? • How does team relate to organizational goals? Do members understand this relationship? • How well are team resources employed?	• Individual interviews followed by group meetings to review findings • Short questionnaires • Observation • Brainstorming
Small, Total Organizations (relatively simple & homogeneous)	• Entire organization at one location; relatively simple and homogeneous.	• How do leaders & members see the organization and its goals? • What do they like and dislike about it? • What is their vision for the organization? • What significant external forces impact the organization?	• Questionnaires, such as descriptive adjective, to obtain a quick reading on the culture, "tone," and health of the organization • Interviews & Observation

Table B-5: Developing Organizational/Doctrinal Profiles (cont.)

(French & Bell, 1973)

Focus	Description & Examples	Information Sought	Investigation Methods
Interface or Intergroup Subsystems	• Subsets of the total system that share common problems & responsibilities, such as production & maintenance overlaps, etc.	• How does each subsystem see the other? What is the climate between two groups? What do the members want it to be? • Are goals, subgoals, areas of authority and responsibility clear? • What problems to subsystems have interacting or working together? • How do they collaborate to perform effectively?	• Confrontation meetings between both groups for data gathering and planning. • Organizational mirroring meeting for 3 or more groups • Interviews of each subsystem, followed by meetings to share & discuss findings • Observation
Roles	• Set of behaviors associated with individual position • Example: leadership roles, functional responsibilities, communication behaviors	• What is the relationship between the role and the subsystem and organizational goals? • Is the role defined adequately? • Should role behaviors be changed? • What is the fit between typical individual (training, experience, etc.) and role?	• Observations • Interviews • Role analysis techniques • Organizational human resource planning definitions

Table B-6a: Design Guidance Associated with Organizational Response

(Andriole & Ehrhart, 1990; Meister, 1991; Rasmussen & Vicente, 1989)

Flexible **Decentralized** **Democratic** **Informal**	**Rigid** **Centralized** **Authoritarian** **Formal**
Triggering Situational Context • Severely stochastic to indeterminate, highly dynamic, high threat environment; short decision horizon • Crisis	**Triggering Situational Context** • Determinate to moderately stochastic, reduced threat, relatively static environment; longer decision horizon • Routine, standard operational procedures
Effects • Respond to problems arising in sphere of responsibility with only general guidance from superior authority • Rapid responses; adaptable to novel situations • Decision makers gain experiences to develop a wide range of creative responses. • Focus on immediate problem may result in satisficing response not meeting organizational objectives • Communication delays may impair information gathering and decision implementation • Potential for crisis decision making behaviors (Table B-6b)	**Effects** • Respond to problems arising in sphere of responsibility according to specific guidance from superior authority • May be ill-prepared for sudden shift in environment • Decision makers have little opportunity to develop a wide range of responses. • High degree of control and consistency across all organizational levels ensures meeting objectives • Communication delays between levels of hierarchy lengthen time between decision and action
Design Guidance • Optimize to provide local decision maker most accurate, relevant information and technological means to combine and interpret abstract/symbolic information • Provide doctrinal/procedural overview displays to support interpretation of and effective response to novel or rare events • Provide organizational objectives or goal-based overview displays to prevent cognitive "tunnel vision"	**Design Guidance** • Optimize for faster communication to minimize authorization delays, etc. • Make explicit all constraints/guidance from superiors in boundary displays, thresholds, and other conformance guide representations • Display structural information (i.e., functional cause & effect relationships) to aid development of mental models and support wider knowledge of response options

Table B-6b: Organizational Decision Making Behaviors in Crisis Situations*

(Heimreich, 1988; Hermann, 1972; Janis, 1989; Klein *et al*, 1992; Pew, 1988)

Leadership & Authority	Communication	Decision Making
• Active decision makers reduced to a core team • Commander's attitudes toward rank and authority critical determinant of subordinate's willingness to raise issues appearing to challenge prevailing hypothesis • Weak or inexperienced leaders may be influenced by subordinates to make incorrect decisions	• Increase in communication with relevant internal and external agencies • Increased intra-team communication may lead to general air of confusion (and potentially panic) and increase the impulse to action	• Stress associated with shorter decision horizons results in general narrowing of perceptual focus ("tunnel vision") or issue fixation, rendering decision maker less capable of dealing with multiple stimuli/issues: - Number of information sources used in situation assessment decreases - Number of alternative courses of action considered decreases • Failure to critique the micro-decisions which aggregate to central decision -- related to general decrease in assumption testing • Increased frequency of action -- decision makers feel "impelled" to take some action

* Hermann (1972) defines a *crisis* as a situation which

1. threatens one or more important goals of the organization,
2. permits only a very short decision time before situation changes significantly, and
3. involves novel or unanticipated events which surprise the decision makers.

Table B-7: User's Domain Knowledge: Impacts & CSE Design Guidance

(Andriole, 1986; Meister, 1991; Senders & Moray, 1991)

User's Domain Knowledge

Low	Moderate	High
Characteristics	**Characteristics**	**Characteristics**
• Limited, fragmented models of domain • Very limited ability to recognize prototypical situations • Very limited ability to interpret novel situations • Very limited framework for structuring long- or short-term goals	• Situation-oriented models of domain • Recognize some prototypical situations • Uses analytical reasoning in response to novel situations • Goal structuring primarily defined by learned procedures & situational models	• Wholistic models of domain • Rapidly recognizes prototypical situations • Intuitively interprets novel situations • Goal structuring defined by robust framework based on doctrine & wholistic domain models
Potential Errors	**Potential Errors**	**Potential Errors**
• May not recognize critical situational cues • Limited ability to reason about cues • Lack of confidence may result in slowed response & reluctance to commit to action • Novel situations may induce confusion and error • Limited goal framework increases probability of errors of intent	• May misinterpret situational cues due to limitations of mental models or fixation on most available situational models • Limited ability to resolve conflicts between situational mental models • May fail to recognize the degree & impacts of uncertainty in situational cues	• May not be consistent in combining situational cues • Competition between mental models may trigger availability bias • Over-confidence in situational interpretation
Design Guidance	**Design Guidance**	**Design Guidance**
• Displays formatted as accepted domain models to present situational information in context & map causal relationships • Constraints, supports & reminders to guide domain understanding & increase confidence in situation assessment • Templates of prototypical domain constructs with relevant cues highlighted to assist in comparisons and developing responses in novel situations	• Displays formatted as accepted domain models to present situational information in context & map causal relationships • Support the construction of more robust mental models with option to view deeper levels of explanation • Displays to make explicit the sources and extent of domain uncertainty	• Option to use domain model displays or customize displays & interaction routines to match their mental models • Support the continued development of mental models with option to view deeper levels of explanation • Displays to make explicit the sources and extent of domain uncertainty

Table B-8: User's Functional Task Knowledge: Impacts & CSE Design Guidance

(Andriole, 1986; Meister, 1991; Senders & Moray, 1991)

User's Functional Task Knowledge

Low	Moderate	High
Characteristics • Lack of knowledge experienced as reduced tolerance to workload • Inability to distinguish between relevant and irrelevant information • Inability to generate and evaluate an adequate response	**Characteristics** • Task performance characterized by facility with learned procedures • Moderate ability to trade off performance quality to maintain reasonable workload • Knowledge base for functional tasks is adequate for all routine operations and some novel situations	**Characteristics** • Flexible, intuitive task performance • Rapidly recognizes prototypical situations • Intuitively interprets novel situations • Goal structuring defined by robust framework based on doctrine & wholistic domain models
Potential Errors • Lack of experience leads to inability to maintain performance quality under increased task workload • Time lost reviewing irrelevant information or inappropriate options • Response generation capabilities limited; limited ability to evaluate options • Difficulty prioritizing tasks	**Potential Errors** • Misapplication of learned procedure may result in inappropriate response • Fixation on task features that match stored (especially readily available) schema may prevent decision maker from correctly diagnosis situation • In high information volume situations, may not have adequate schema to distinguish relevant information	**Potential Errors** • Overconfidence in correctness of response • Insensitivity to potential aggregated errors in subtasks (microdecisions) performed in multistage decisions • Failure to revise decision when situation changes • Tendency to think in causal sequences rather than network of contributing causes and consequences of action
Design Guidance • Design guidance from "User Domain Knowledge" (above) applies with a focus on specific functional tasks • Allow user to query constraints & affordances • Adaptive "intelligent" decision aids may be appropriate to filter displays and propose options to decision maker • Organizational structures can provide the same kinds of error trapping, error flagging, and redundancy afforded in machine design	**Design Guidance** • Make task constraints and affordances visible • Provide goal- or decision-oriented displays to focus attention on relevant information • Provide explicit information on the potential effects of subtask uncertainty • Constraints (for error control) & the option to use supports & reminders during situation assessment	**Design Guidance** • Provide option to use goal-oriented displays or customize displays & interaction routines to match their mental models • Constraints (for error control) & the option to use supports & reminders during situation assessment • Displays to make explicit the sources and extent of uncertainty in key variables

Table B-9: User's System Interaction Knowledge: Impacts & CSE Design Guidance

(Andriole, 1986; Ehrhart, 1990; Norman, 1988; Senders & Moray, 1991)

User's System Interaction Knowledge

Low	Moderate	High
Characteristics • Novice or casual user • Limited, partial knowledge of system operation • User has insufficient mental model of system and may be confused by errors • Increase workload will result in greatly impaired performance	**Characteristics** • *Competent user* • Understands operation of all commonly used system features • Operates interface to accomplish tasks • Successfully learns from operational errors	**Characteristics** • Master; "power" user • Strong, accurate mental model of the system relationships between self, machine and tasks to perform • Fluid operation of interface; interface tasks are transparent -- user is directly involved in functional tasks
Potential Errors • Variety of errors of intention (mistakes) due to inexperience • Slips (right intention - incorrect action) • Casual user will forget procedures	**Potential Errors** • Mistakes due to incomplete or flawed mental models; repeating errors due to incorrectly learned sequence • Mode errors based on incorrect assumptions about current system state • Slips	**Potential Errors** • Ability to bypass some operational sequences may result in unintended action • Illogical design will still confound the expert user • Slips
Design Guidance • Provide overview screens to help user develop a mental model of the system resources available and understand where they are in a process • Make system state explicit • Build in system capability to prevent "fatal" error, alert user to nature of error and response options • Make current options (affordances) visible • Make use of natural or domain knowledge in the interaction symbology to allow the user to interact with the task in the most familiar terms	**Design Guidance** • System state, available options and similar information should be visible or available on demand • Provide option to shift to overview displays for orientation • Minimize the use of similar interaction sequences varying in effect in different operational modes • Facilitate error recovery through "undo" commands • Design levels of help to allow user to select the depth of information desired	**Design Guidance** • Basic redundancy and error tolerant design guidelines apply • Allow user to tailor interface to optimize for best performance

Table B-10: General Characteristics of Key Variables: Inputs, Outputs & Feedback

(Ehrhart, 1993; Meister, 1991)

Variable	Description	HCI Design Issues
Modality	written, spoken, visual, aural	structure; presentation; highlighting; manipulation
Structure	quantitative, qualitative; structured, unstructured	abstraction level; manipulation
Content	information provided	structure & organization
Intensity	strong, weak; detectability	alerts; presentation; highlighting; modality (above)
Immediacy	immediate, delayed, constant	association with source; presentation; highlighting
Volume	too much, too little, appropriate (relative to problem)	task allocation; filtering; aggregation & abstraction; highlighting
Duration	short, long, continuous	task allocation; detection; presentation; highlighting
Uniqueness	presence or absence of other associated information	presentation structure & organization; abstraction; highlighting
Specificity	specific or general with respect to content or source	structure & organization; abstraction; presentation;
Consistency	with other related information	presentation structure & organization; abstraction; highlighting
Source Location	internal , external; hierarchical level	presentation; highlighting
Linearity	linear, non-linear (relative to source)	presentation structure & organization; abstraction; highlighting
Dimensionality	uni-dimensional, multi-dimensional	presentation structure & organization; abstraction; highlighting; manipulation
Reference	organization, unit; external	presentation structure & organization
Expectation	consistent / inconsistent with system expectations	structure & organization; alerts; presentation; highlighting

Table B-11a: Functional Task Characteristics: Output

(Fleishman & Quaintance, 1984; Gardiner & Christie, 1989; Meister, 1991)

Task Characteristic	Description	Scale			Implications for HCI Design
		Low	Mid	High	
Number of Output Units	Number of units produced during time period; Throughput or task volume	One	Moderate number	As many as possible	• Key issues: task allocation; short-term memory; attention • Display and interaction design (e.g., screen refresh, item selection mechanism, etc.) must be appropriate for speed requirements • Processing requirements may preclude certain display modes, depending upon precision, etc.
Number of Elements per Unit	Number of component parts per output unit	One	Several	Many	• Key issue: identification of appropriate level of detail • In multi-component tasks user's may benefit from displays which organize component information for faster processing (i.e., data tables) depending upon level of detail required • Task complexity factor (see also Procedure/Subtask Characteristics in Table B-11c below)
Duration Output Unit Maintained	Duration output unit must be maintained/continued	Minimal	Moderate	As long as possible	• Key issues: impacts on attentional or short-term memory resources; task allocation design (see Output Workload)
Output Workload	Function of number of units and duration output is maintained	Low	Moderate	High	• Key issue: impacts of extended vigilance on attentional or short-term memory resources • Higher workload situations may require adaptive re-allocation of tasks to relieve human user

Table B-11b: Functional Task Characteristics: Response Characteristics

(Fleishman & Quaintance, 1984; Meister, 1991)

Task Characteristic	**Description**	**Scale**			**Implications for HCI Design**
		Low	Mid	High	
Goal Attainment Difficulty	Function of the number of elements per output unit and workload	Both Dimensions Low	Both or Average Moderate	Both Dimensions High	• Key issue: task complexity factor, impacts on user frustration and motivation levels • Very high ratings may indicate tasks out of the range of human performance • Low ratings have implications for maintaining user attention / interest
Response Precision	Most precise response required in any output unit	Imprecise	Moderately Precise	Extremely Precise	• Key issues: impacts information display precision and response input mechanism • Relates to issues of aggregation and abstraction in information presentation • Impacts interaction modes & feedback depending upon throughput requirements
Response Frequency	Responses required per unit time	Low (1 - Few)	Relatively Moderate	Very High or Continuous	• Key issues: demand on attentional resources; response input mechanism; task allocation strategies • Very high or continuous response rates may exceed human performance • Feedback design should accommodate response rate through aggregation (e.g., level indicator vs. discrete feedback)
Simultaneity of subtasks	Number of subtasks that must be performed simultaneously	One	At Least Two	Several	• Key issues: demand on attentional focus; response input mechanism; response feedback design • More than one simultaneous subtask may suggest need for dynamic task reallocation under higher workload conditions

Table B-11c: Functional Task Characteristics: Procedure / Subtask Characteristics

(Fleishman & Quaintance, 1984; Meister, 1991)

Task Characteristic	Description	Scale			Implications for HCI Design
		Low	Mid	High	
Number of Procedural Steps	Number of responses or subtasks required to produce one output unit	Small (≤ 3)	Medium	Very Large	• Key issues: task complexity factor; impacts on short-term memory • Large number of procedural steps may indicate need for user support in the form of prompts or reminders
Dependency of Procedural Steps	Interdependence of steps (temporal order, etc.)	None to Low	No more than 50%	Very High	• Key issues: task complexity factor; impacts on short-term memory • User may benefit from causal diagrams and information presentation techniques indicating task status
Adherence to Procedures Required	Degree of adherence to set procedure required (rigidity vs. flexibility)	Great flexibility tolerated	some flexibility tolerated	Strict adherence required	• Key issues: impacts on level of autonomous control extended to user; attention requirements • Where strict adherence is required, dialogue may be machine-driven to conform to procedural rules • Feedback should indicate whether task is "on track"
Procedural Complexity	Function of number of subtasks and dependencies between subtasks	Few steps - little or no dependence	Several steps - some dependence	Many steps - all dependent	• Key issues: task complexity factor; impacts on short-term memory • High procedural complexity suggests need for information display to support user understanding of dependencies and progress through task in terms of subtasks completion, etc.

Table B-11d: Functional Task Characteristics: Input Characteristics

(Fleishman & Quaintance, 1984; Gardiner & Christie, 1989; Meister, 1991)

Task Characteristic	Description	Scale			Implications for HCI Design
		Low	Mid	High	
Stimulus Variability	Predictability of stimuli attributes over task time	Unchanging	Varies in known pattern	Random	• <u>Key issues</u>: stimulus detection and identification; long-term memory • Low - Analogical supports for categorizing random stimuli • Mid - memory aids & template displays to assist in identifying known patterns
Stimulus Duration	Duration of stimulus relative to task time	Stimulus ends before response initiation	Stimulus remains until changed by response	Stimulus may remain indefinitely	• <u>Key issues</u>: impacts on attention & short-term memory requirements • Low stimulus duration may require stimulus storage and re-display; change in stimulus (e.g., sampled variables) may require display to indicate trends • Duration impacts feedback requirements (i.e., status of responses to visible or invisible stimuli)
Occurrence Regularity	Predictability of stimulus occurrence	Very irregular to random	Varies in known pattern	Regular or constant	• <u>Key issues</u>: stimulus detection; attention • Low - may require alert mechanism to notify user of stimulus occurrence • Mid - may benefit from trend displays and templates to assist in recall
Decision Maker's Control of Stimulus	Degree of control decision maker has over occurrence or relevance of stimulus	No Control	Some Control	Total Control	• <u>Key issues</u>: stimulus detection and identification; workload & frustration • Lower control suggests requirement to alert user to stimulus occurrence • Filtering displays may reduce the effects of irrelevant stimuli & reduce "false sensations," confusion, and associated workload

Table B-11e: Functional Task Characteristics: Feedback

(Boff & Lincoln, 1988; Fleishman & Quaintance, 1984; Meister, 1991; Vicente & Rasmussen, 1992)

Task Characteristic	**Description**	**Scale**			**Implications for HCI Design**
		Low	Mid	High	
Decision Maker's Control of Response Lag	Degree of control decision maker has over how fast they must respond	None - must respond immediately	Some - respond within time range	Full - responds at own discretion	• <u>Key issues</u>: task allocation strategies • Event-driven response may require dynamic adaptation to automated response • Response within set time range suggests elapsed time displays, etc.
Feedback Lag	Speed of feedback on decision maker action [Feedback is also an input (see Stimulus above)]	No feedback received	Feedback delayed	Immediate feedback	• <u>Key issues</u>: attention; impacts on short-term and long-term memory • Experiential learning requires feedback • Delayed or no feedback may require display aids to maintain user understanding of causal relationships and reduce demands on short-term memory • Lack of feedback has negative impact on decision maker's ability to task knowledge base; must be addressed through training and/or information presentation design
Reaction Time / Feedback Lag	Ratio of decision maker's reaction time to feedback lag	Reaction time < feedback lag	Reaction time = feedback lag	Reaction time > feedback lag	• <u>Key issues</u>: impact of feedback on performance quality • Fast response with delayed feedback may result in over-correction (mis-interpretation of feedback reference) or developing of inaccurate causal models
Number of Choice Subtasks	Number of subtasks involving decision maker choice based on feedback or outcome of last response	< 25%	~ 50%	> 75%	• <u>Key issues</u>: impact of feedback on performance quality; short-term memory • Relates to task complexity due to dependencies • Suggests greater potential for task variability due to branching effect

Table B-12a: Decision Task Characteristics: Stimulus

(Andriole & Adelman, 1989; Andriole & Ehrhart, 1990, 1993; Cohen, 1985; Gardiner & Christie, 1989; Meister, 1991; Vicente & Rasmussen, 1992; Wohl, 1981)

Decision Task Characteristic	Description	Scale			Implications for HCI Design
		Low	Mid	High	
Attentional Requirements	Level of vigilance required in task performance (i.e., monitoring workload)	Little or no active monitoring required	Monitoring at set or random intervals	Continual monitoring required	• Key issues: impacts on attention; short-term memory; task allocation, and workload • Low monitoring requirement may result in poor situational awareness when stimulus occurs; highlight changes • Decision makers monitoring at intervals may benefit from memory supports such as trend displays or reminders • Requirement to maintain vigilance over long periods may result in fatigue and variation in attentional focus; continual monitoring is best done by machines with alerts to decision maker
Detection Difficulty	Degree of difficulty in detecting or discerning stimuli	None to very little; immediately detectable	Some difficulty - time lag may occur	Very difficult - masked by other stimuli or deception	• Key issues: stimulus alerts and display • Trend displays may aid where detection is difficult due to time lag • Filters and pattern templates can assist in identifying relevant stimuli and detecting masked stimuli

Table B-12a: Decision Task Characteristics: Stimulus (cont.)

(Andriole & Adelman, 1989; Cohen, 1985; Meister, 1991; Vicente & Rasmussen, 1992; Wohl, 1981)

Decision Task Characteristic	Description	Scale			Implications for HCI Design
		Low	Mid	High	
Level of Abstraction	Level of detail in stimuli and decision cues	Highly detailed, not summarized (Signals)	Moderate detail & aggregation (Signs)	Low detail, highly aggregated (Symbols)	• Key issues: information display and interpretation; error characteristics • Stimuli occurring as signals may be aggregated for interpretation as signs • Misinterpretation of signs can result in incorrect response due to fixation or inaccurate recall • Interpretation of symbolic, abstract information benefits from reminders of the underlying causal structures • Symbolic representation of information must adequately represent all relevant dimensions ("law of requisite variety)
Qualitative vs. Quantitative	Extent to which stimuli are qualitative vs. quantitative	Little or no qualitative variables	Mixture of qualitative and quantitative	Highly qualitative; cannot be legitimately quantified	• Key issues: information display and interpretation • Interpretation of highly quantitative data may benefit from graphic presentations and aggregation, guided by level of detail requirements • Interpretation of mixed or highly qualitative information may be aided with templates and models that support recall of learned patterns or relevant analogs from long-term memory

Table B-12a: Decision Task Characteristics: Stimulus (cont.)

(Andriole & Adelman, 1989; Cohen, 1985; Gardiner & Christie, 1989; Meister, 1991; Reason, 1990; Wohl, 1981)

Decision Task Characteristic	Description	Scale			Implications for HCI Design
		Low	Mid	High	
Memory Requirements	Rate and volume of incoming stimuli/information	Easily managed with human memory	Manageable by highly motivated experts	Exceeds human ability to absorb or manipulate	• Key issues: impacts on short-term memory • Memory-intensive tasks may be partially or fully allocated to machine; particularly low level data integration, computation, etc. • Present information at the highest level of abstraction suitable for the decision task to reduce memory demands & maintain situational awareness
Reliability & Representativeness	Extent to which decision variables are understood and can be used to reliably assess situation	Automatic; Clear indicator	Generally understood, representative, & reliable	Not well-understood or unreliable "unknown unknowns"	• Key issues: misperceptions; judgment & reasoning errors • Misperception due to incomplete or ambiguous information may result in - focus on irrelevant information; - selection and/or fixation on incorrect explanation schema or solution - incorrect interpretation of cues; or - insensitivity to missing information • Presence of "unknown unknowns" (relevant, but hidden cues) may result in developing faulty causal schema • Displays of system models or goal states may aid - problem identification - defining causal relationships - identifying missing information - interpreting ambiguous cues - reducing over-confidence in decisions based on uncertain information

Table B-12b: Decision Task Characteristics: Hypothesis

(Andriole & Adelman, 1989; Cohen, 1985; Meister, 1991; Reason, 1990; Wohl, 1981)

Decision Task Characteristic	Description	Scale			Implications for HCI Design
		Low	Mid	High	
Situation novelty	Degree of novelty in situation	Little or none; routine	Somewhat familiar	Very unfamiliar	• Key issues: misperceptions; long-term memory, judgment & reasoning errors • Routine situations may be handled with procedural reasoning or automated to reduce workload • Decision makers use analogies to previous experience and scenarios to generate and test hypotheses • Displays of analogous information patterns or templates can aid in categorizing somewhat familiar situations to allow procedural response • Formal reasoning required in novel situations may suffer from - failure to consider processes across time - tendency toward thinking in linear (causal series) rather non-linear sequences (causal nets)
Number of Possible Hypotheses	Number of possible explanations for stimuli	Few; well-bounded	Moderate; semi-bounded	Very many; unbounded	• Key issues: misperception; memory, judgment & reasoning errors • Well-bounded situations may be candidates for rule-based automated support • In less well-bound situations with many possible hypotheses, decision makers benefit from displays that aid in structuring information to reduce the number of hypotheses actively considered • Models presented at higher levels of abstraction can aid in identifying aberration from desired state.

Table B-12b: Decision Task Characteristics: Hypothesis (cont.)

(Andriole & Adelman, 1989; Andriole & Ehrhart, 1990, 1993; Klein *et al*, 1992; Meister, 1991; Wohl, 1981)

Decision Task Characteristic	Description	Scale			Implications for HCI Design
		Low	Mid	High	
Decision Horizon	Speed required in situation assessment	Planning & forecasting	Time critical	Real-time	• Key issues: attention, memory, workload, judgment & reasoning errors • As decision horizon shortens, several factors become more critical: - experience level - attention focus & vigilance level - feedback speed • Errors in shorter decision horizons due to trading off performance accuracy to meet response speed requirements • Display and interaction design for short decision horizons should - highlight relevant information & filter out irrelevant information to facilitate "at a glance" processing - optimize task allocation to reduce decision maker's "off-line" tasks • Longer range planning may suffer from delayed feedback; displays should aid in understanding cause/effect relationships
Inferencing Required	Degree of inferencing required to assess situation	None to very little	Some - within set bounds	Extensive	• Key issues: memory, reasoning errors • Higher inferencing required for more stochastic or indeterminate situations • Feedback on actions may also be ambiguous, requiring further inferencing • Multi-dimensional inferencing is extremely memory intensive • Attempts to reduce workload can result in reasoning errors (cognitive biases) • Inferencing involves causal reasoning • Displays should aid in mapping the causal net inferred by decision maker

Table B-12c: Decision Task Characteristics: Option

(Andriole & Adelman, 1989; Gerhardt-Powals, 1992; Klein *et al*, 1992; Meister, 1991; Wohl, 1981)

Decision Task Characteristic	**Description**	**Scale**			**Implications for HCI Design**
		Low	Mid	High	
Number of Possible Options	Number of potential responses to hypothesized situation	Few; well-bounded	Moderate; semi-bounded	Very many; unbounded	• Key issues: attentional focus; memory; information processing; judgment & reasoning errors • Well-bounded situations with narrow response ranges may be candidates for automation or rule-based support • Increased number of feasible responses can cause decision maker to jump from option to option or attempt to over-simplify • In less well-bound situations with many possible responses, decision makers benefit from displays that aid assessing consequences of actions
Tractability	Degree of difficulty involved in evaluating options; may be a function of information volume, problem boundedness, or both.	Highly tractable	Variables difficult, but tractable	Intractable or exceeds available resources	• Key issues: memory, information processing; judgment & reasoning errors • Intractability due to volume may be mediated with machine support • Intractability due to problem boundedness requires validated abstraction models to reduce overall complexity • Workload may be reduced with displays that map relative values of multidimensional outcomes against goals • Causal models, analogs & goal displays can assist in selecting response

Table B-12c: Decision Task Characteristics: Option (cont.)

(Andriole & Adelman, 1989; Klein *et al*, 1992; Meister, 1991; Wohl, 1981)

Decision Task Characteristic	Description	Scale			Implications for HCI Design
		Low	Mid	High	
Goal Variability	Extent to which goals shift and/or conflict	Stable, unconflicted goals	Goals moderately dynamic; predictable conflicts	Goals shift rapidly with potentially significant conflicts	• Key issues: memory; feedback; judgment & reasoning errors • Timeliness of feedback is critical as goals change more rapidly • Shifting goals require re-prioritization and re-evaluation of current options against higher-level goals • Multi-stage decisions may be superseded by events mid-decision • Predictable goal changes may be combined into scenario templates and displayed to user as advance notice or incorporated into a rule-based advisor
Evaluation Difficulty	Degree of difficulty in assessing values of options	Outcome values well-understood & easily determined	Outcome values may be determined with some effort	Outcome values poorly understood or very difficult to determine	• Key issues: feedback • Closely related to boundedness and tractability of problem • Well-understood, easily determined option values trigger SOP responses and may be candidates for automation • Higher levels of evaluation difficulty may result in unacceptable delays in decision making or reluctance to commit to any option -- wait to see what breaks • Difficult evaluation may be supported with tools that allow rapid scoring of options against basic criteria with predetermined or adjustable weighting • Displays may present outcomes or animate projected consequences for comparison

Table B-12c: Decision Task Characteristics: Option (cont.)

(Andriole & Adelman, 1989; Klein *et al*, 1992; Meister, 1991; Wohl, 1981)

Decision Task Characteristic	**Description**	**Scale**			**Implications for HCI Design**
		Low	Mid	High	
Outcome Uncertainty	Degree of uncertainty associated with outcome value(s)	Little or no uncertainty	Some uncertainty, but predictable	Highly uncertain; unpredictable	• Key issues: feedback; inferencing requirements • Greater uncertainty requires more inferencing (see above) • Decisionmaker may be reluctant to commit • Determining the potential effects of decision across complex system may become intractable • Ambiguous and/or delayed feedback can impair dependent decisions and experiential learning • Feedback and successive correction may allow decision maker to adjust for outcome uncertainty if feedback is timely, goals do not change, and there is not a high penalty for an incorrect response

Table B-12d: Decision Task Characteristics: Response

(Andriole & Adelman, 1989; Klein *et al*, 1992; Meister, 1991; Wohl, 1981)

Decision Task Characteristic	**Description**	**Scale**			**Implications for HCI Design**
		Low	Mid	High	
Planning Required	Amount of planning required to implement decision	Little or none; automatic	Manageable with ad hoc or existing plans	Extensive planning or replanning required	• Key issues: memory; reasoning; decision horizon • Extensive planning or replanning may require - reassessment of goals and subgoals - adjustment of control structures • Plans are hypotheses involving assumptions and inferences about causal relationships (see Tables 12b & c) • Information presentation and manipulation must support the goal decomposition and means-end restructuring required
Coordination Required	Extent to which decision maker must coordinate	Coordination within local unit	Coordination within organization	Coordination involves external organizations	• Key issues: memory; organizational structure; decision horizon • Coordination affected by formal and informal organizational structures (see Tables 6a & b) and decision horizon (see Table B-12b) • Coordination tasks are communication-intensive; may effect design tradeoffs • Communication may require re-formatting to match transmission or reception capabilities; transformation can fundamentally change information characteristics (see Table B-10) and affect interpretation • Displays on coordination links assist in ad hoc restructuring

Table B-12d: Decision Task Characteristics: Response (cont.)

(Andriole & Adelman, 1989; Klein *et al*, 1992; Meister, 1991; Wohl, 1981)

Decision Task Characteristic	**Description**	**Scale**			**Implications for HCI Design**
		Low	Mid	High	
Execution Control Requirements	Extent to which decision involves multiple, dependent steps/phases	Direct and/or single-phased	Several phases with limited dependency	Highly dependent multiple phases	• Key issues: memory; organizational structure; decision horizon • Multiple phases increase coordination requirements (see above) • Multiple phases increase difficulty of tracing all possible consequences (ripple effects) of actions taken • Delayed feedback may be associated with wrong phase; resulting confusion may cause over correction • Memory load, fundamental ambiguity and potential for goal shift increases with complexity pr procedural steps (see Tables 11c & 12c) • Provide displays of goal/subgoals and current state(s) of phases and subtasks

Appendix C:

Example Cognitive Maps from the FLEX Case Study

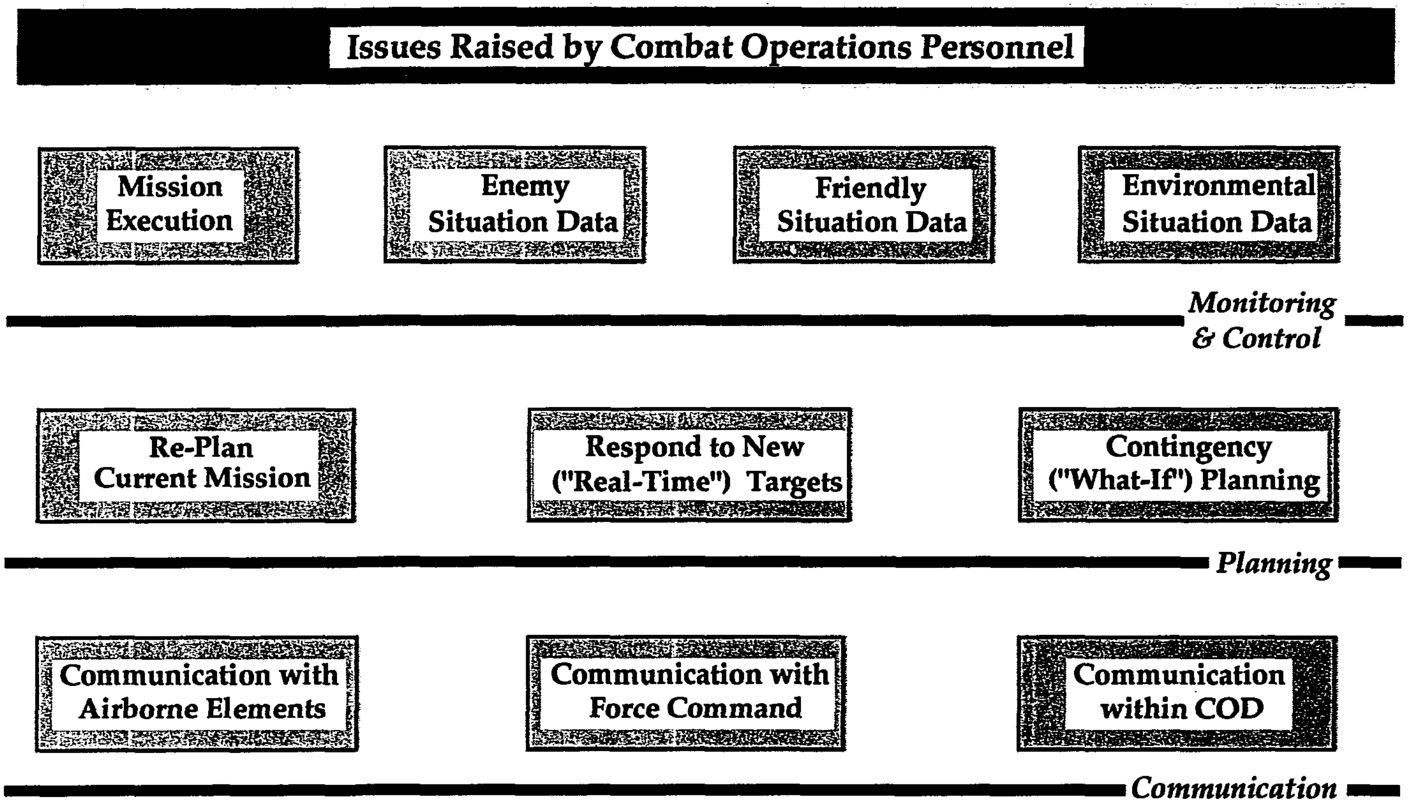


Figure C-1: Combat Operations Division Tasks

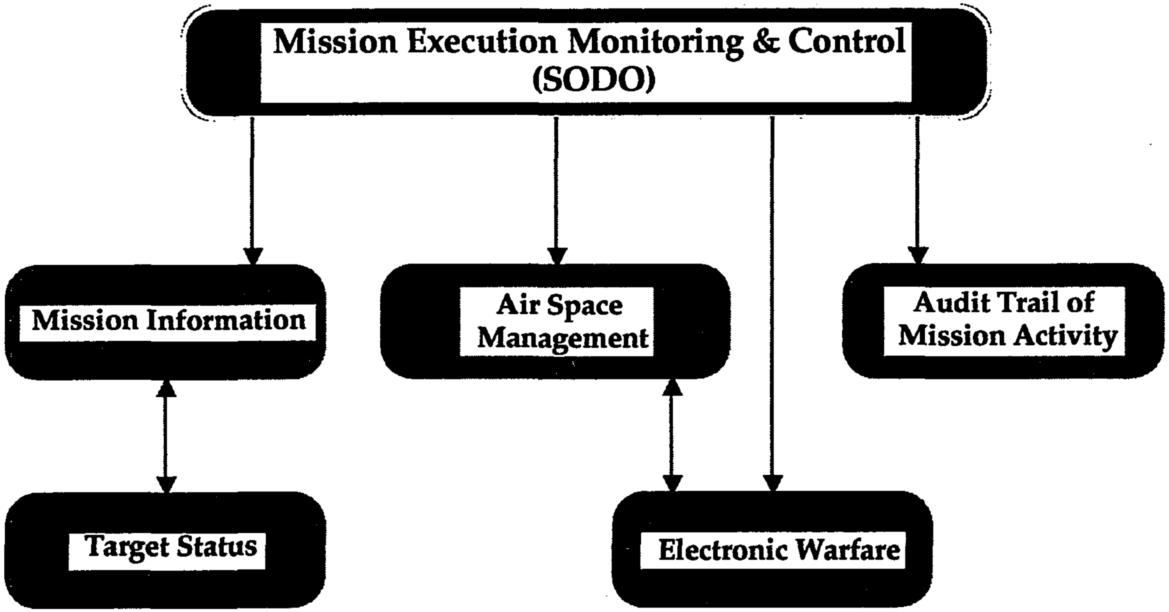


Figure C-2: Mission Execution

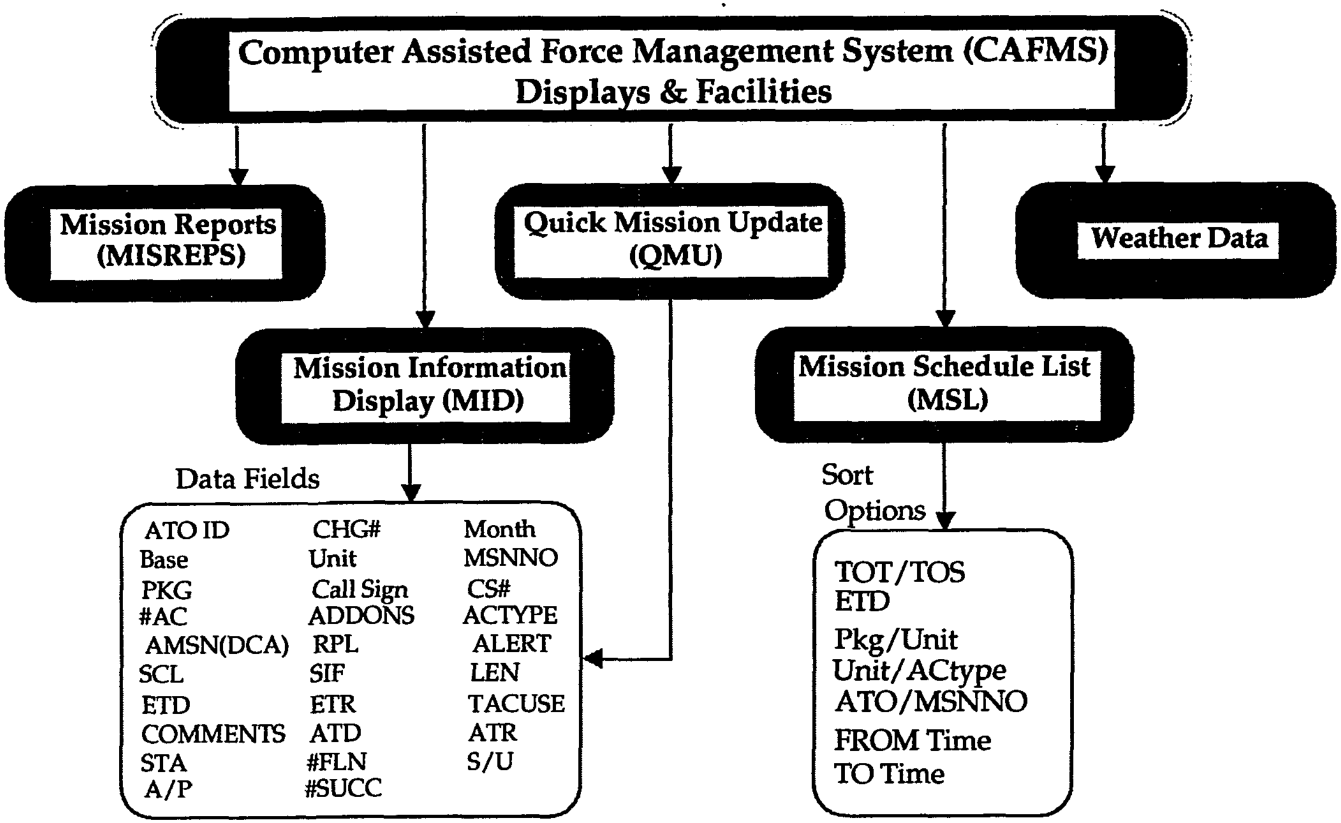


Figure C-3: Mission Information Display

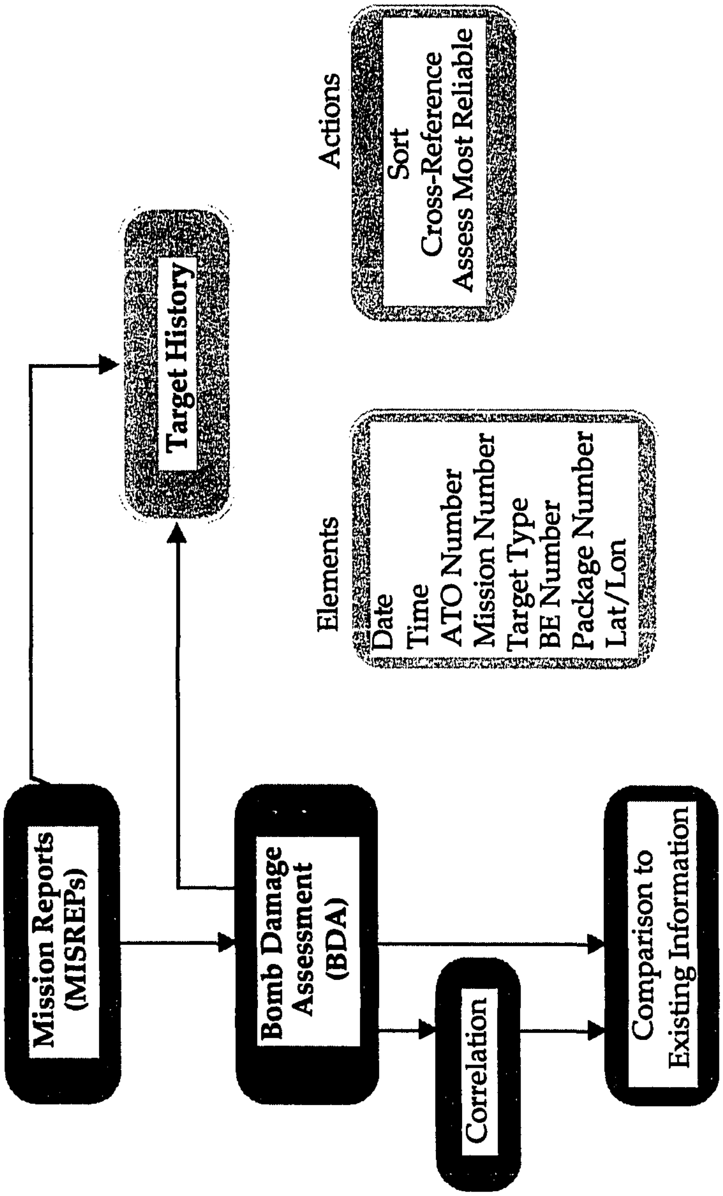


Figure C-4: Target Status

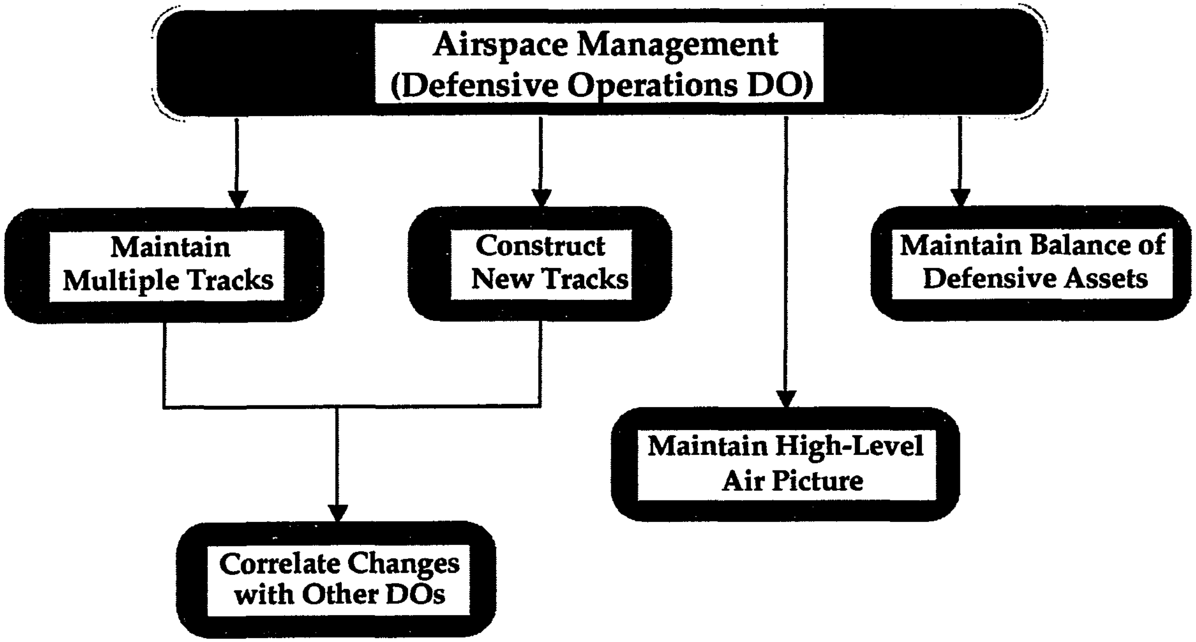


Figure C-5: Airspace Management

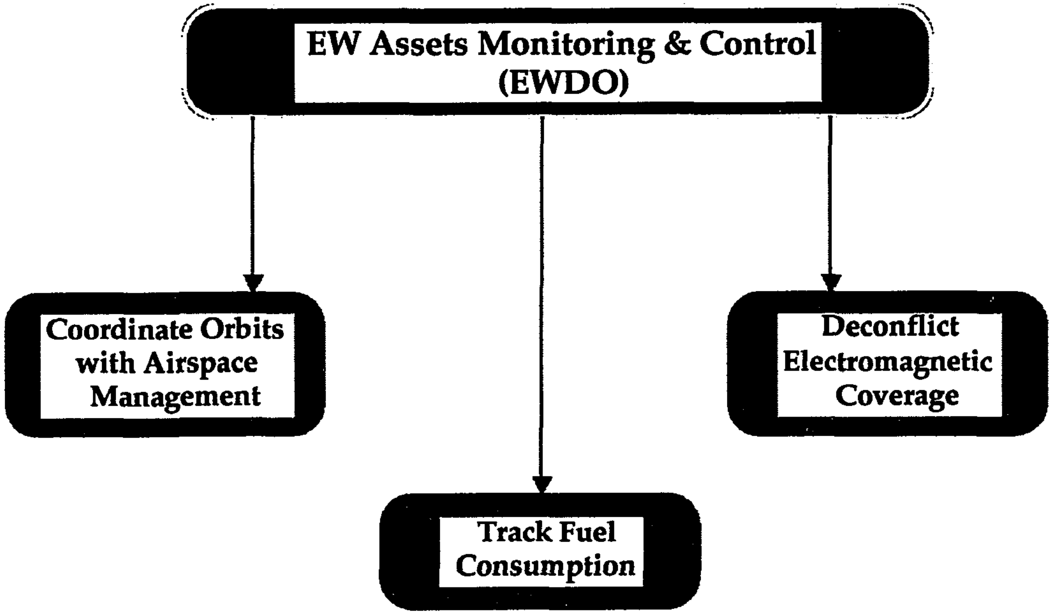


Figure C-6: Electronic Warfare Mission Data

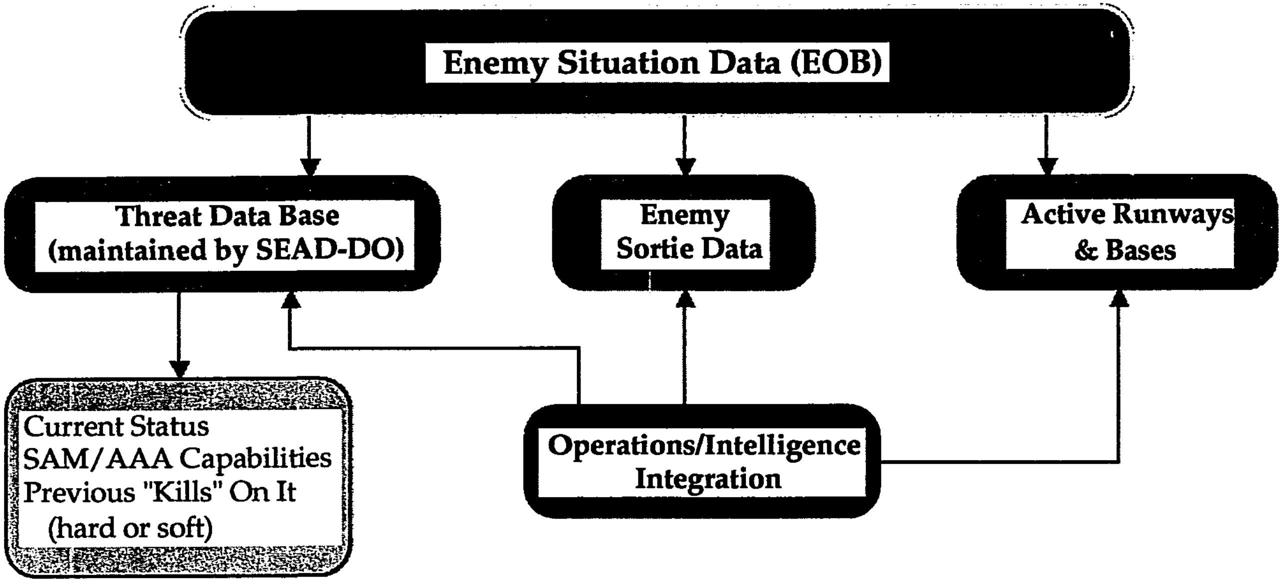


Figure C-7: Enemy Situation Data

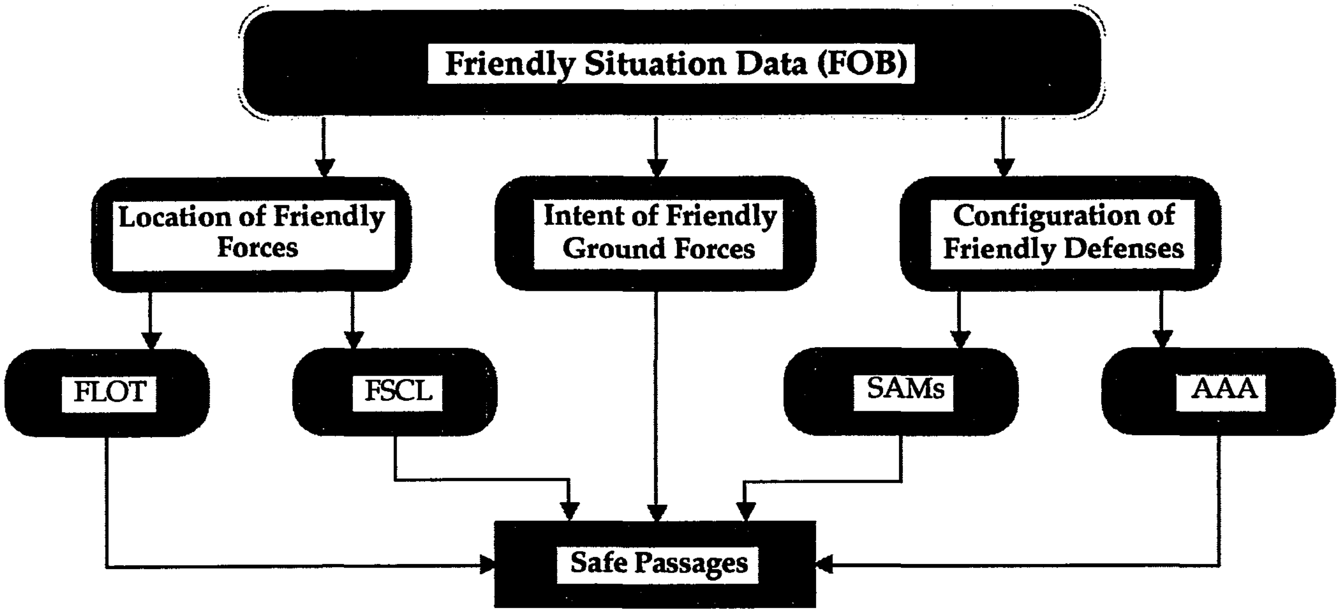


Figure C-8: Friendly Situation Data

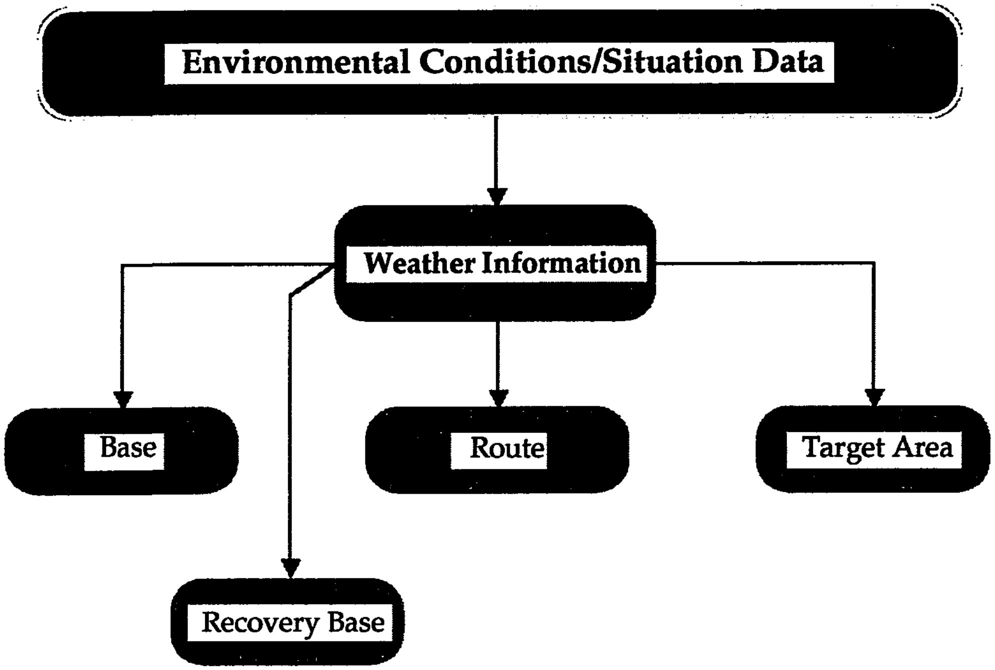


Figure C-9: Environmental Conditions Data

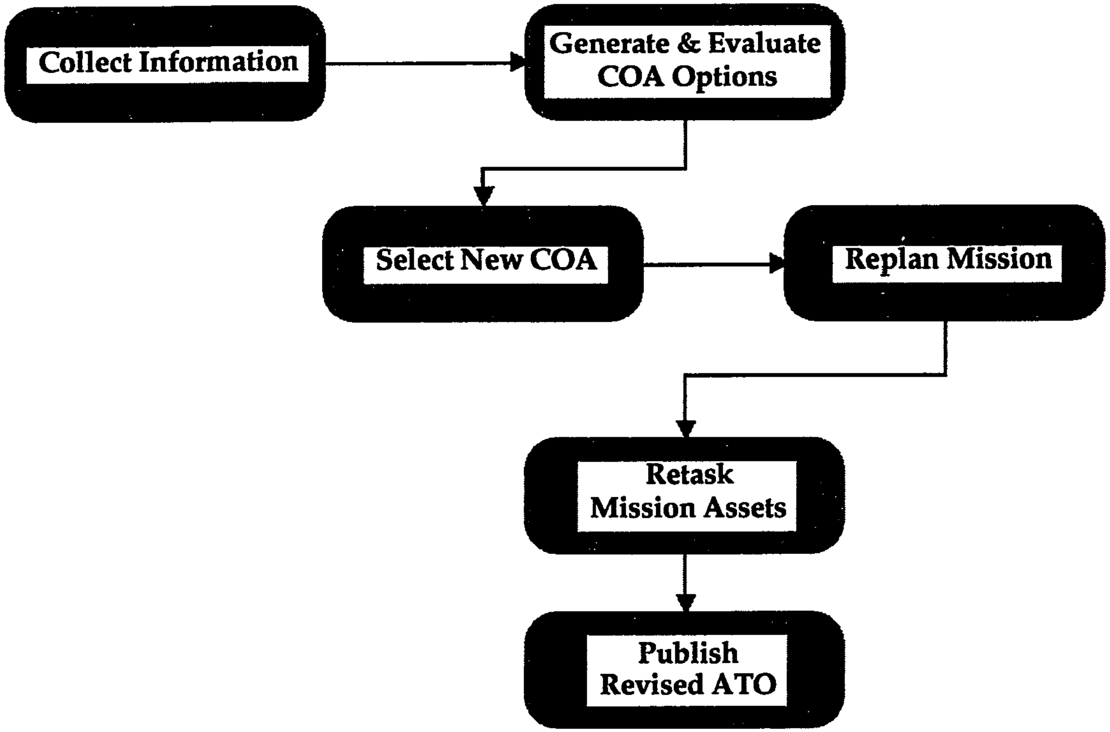


Figure C-10: Re-Plan Current Mission

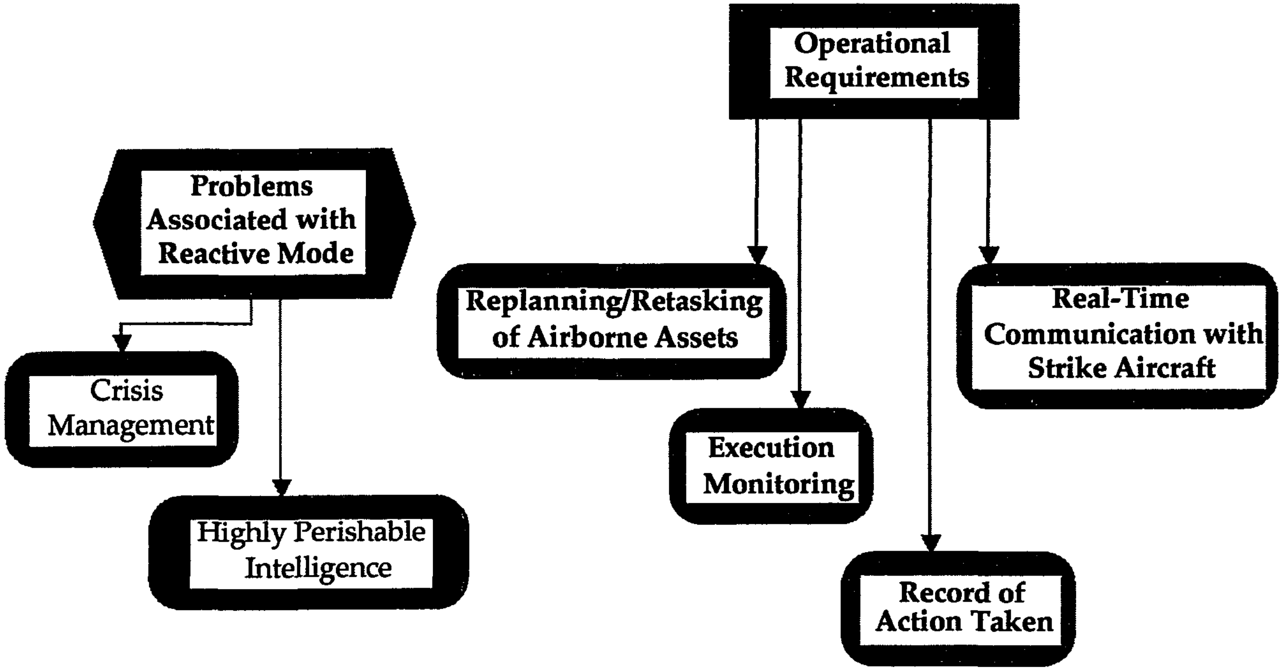


Figure C-11: Response to Real-Time ("Pop-Up") Threats

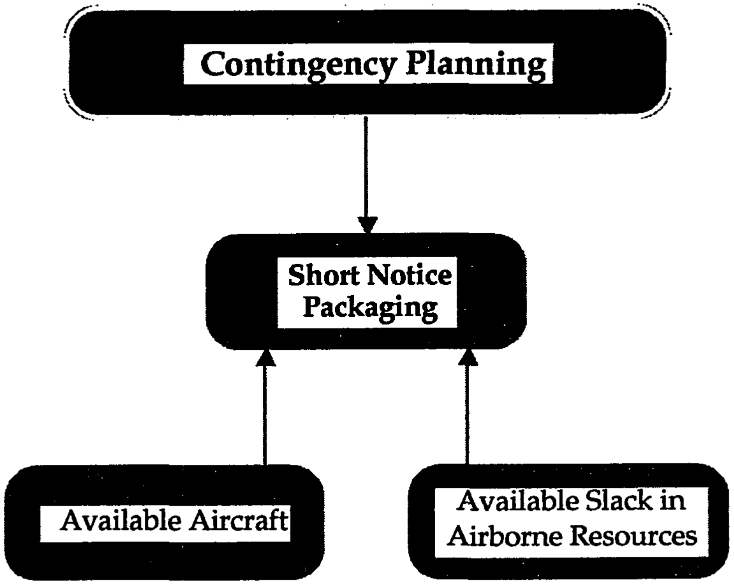


Figure C-12: Contingency Planning

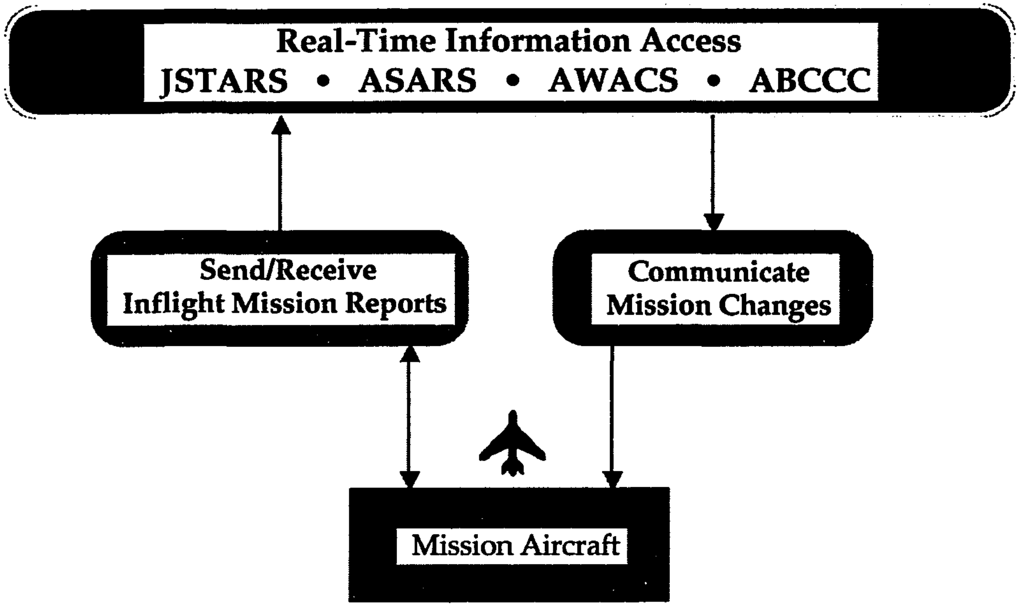


Figure C-13: Airborne Elements

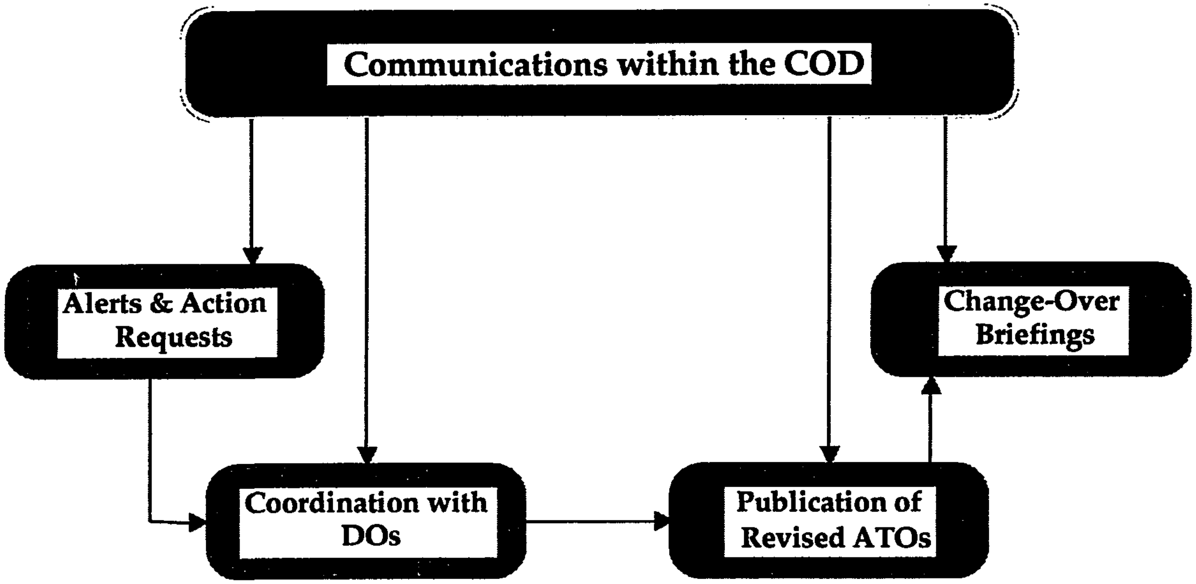


Figure C-14: Communications within the Combat Operations Division

Example Contents

- Alert Status
- Weather Update & Predictions
- Intelligence Update
- Army GLO (FLOT, FEBA, etc.)
- Ops Summary (# sorties, summary data)
- Ops Plans
  - OCA
  - AI
  - EW
  - RECCE
- Resources
- FLOT Crossing Ops
- Offensive Summary

- ASACS Status & Summary
- SAM s
  - Status (# kills)
  - Coverage of the Day
- (plotted on map)
  - Summary
- Fighters
  - Resources
  - Employment
  - Summary
- Air Defense Fighter s
  - Resources
  - Summary

Figure C-15: Change-Over Briefing

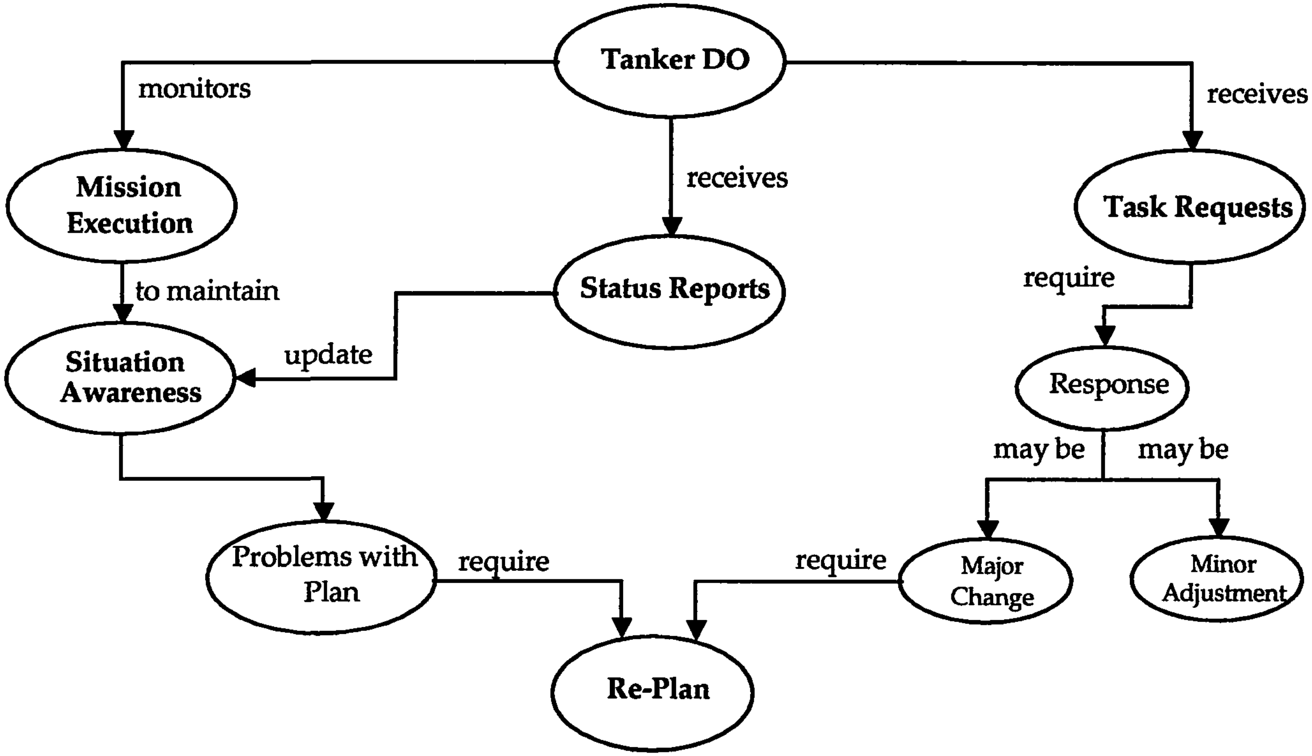


Figure C-16: Tanker Duty Officer Tasks

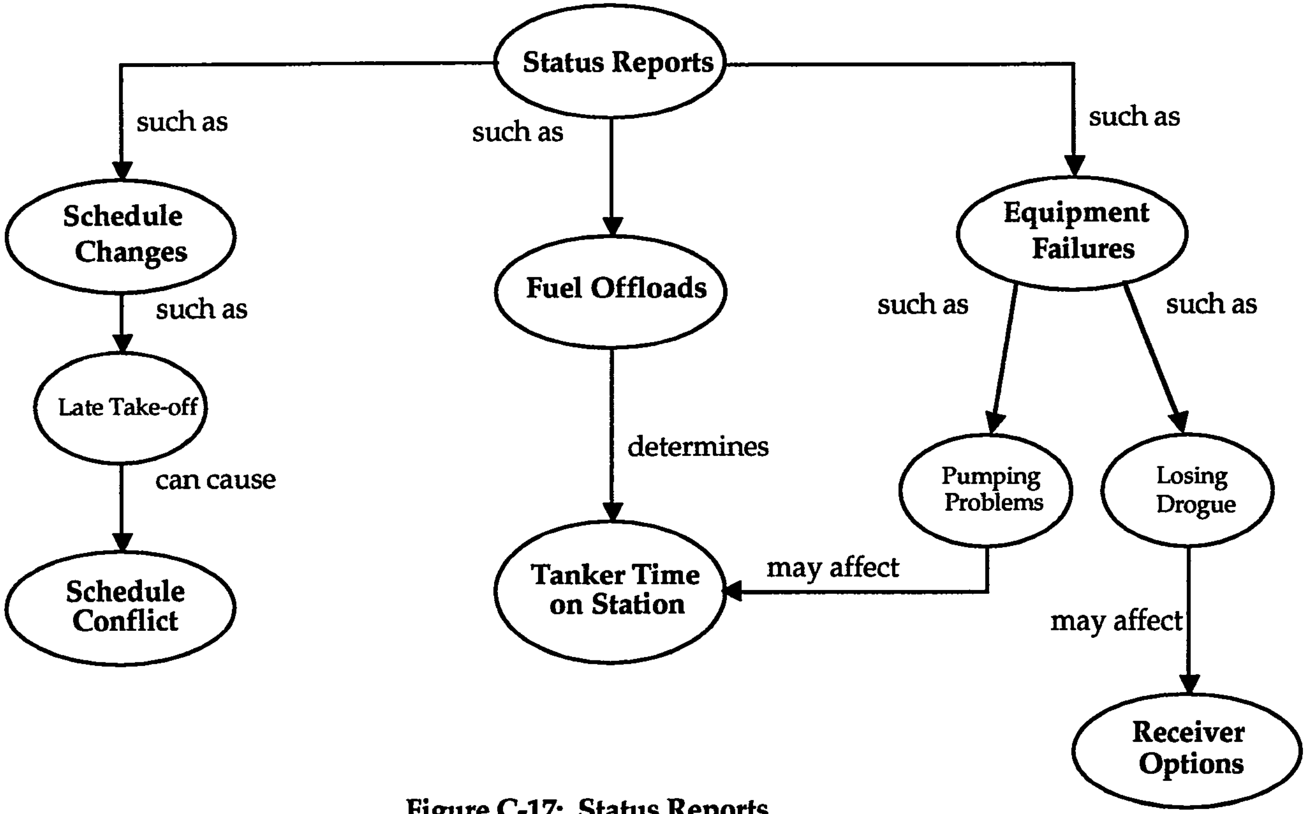


Figure C-17: Status Reports

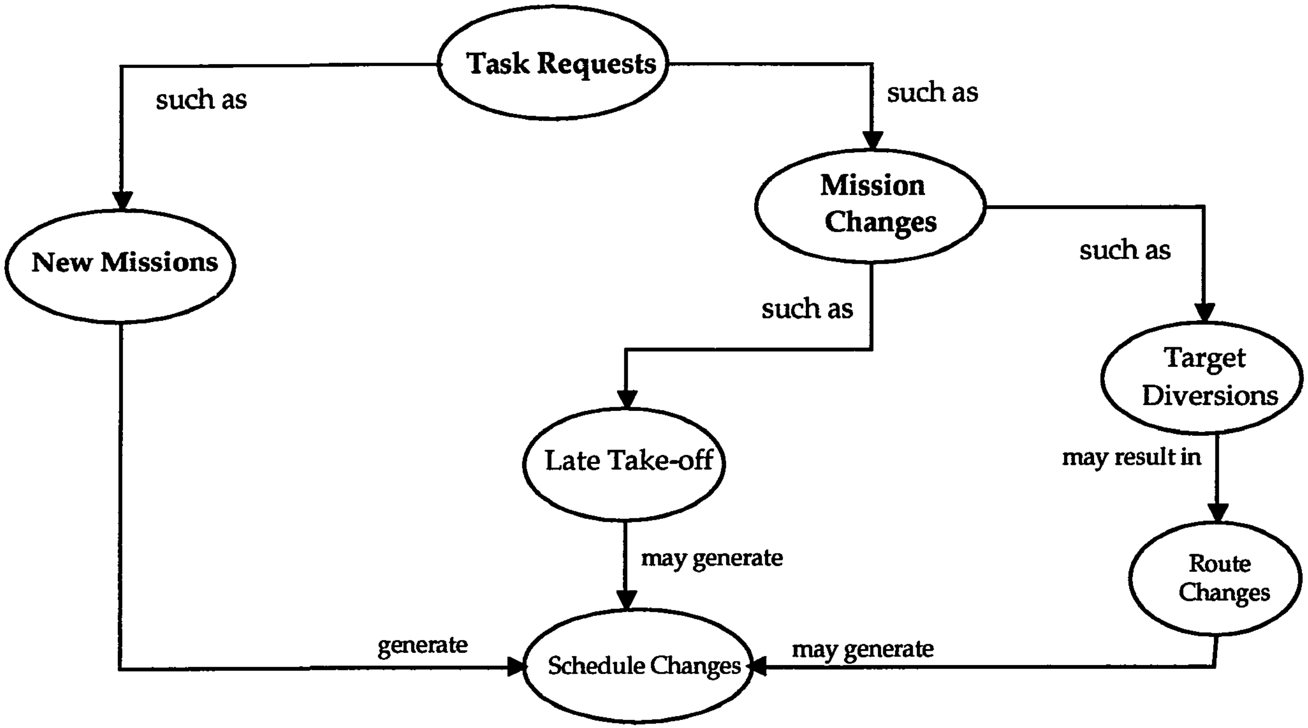


Figure C-18: Task Requests

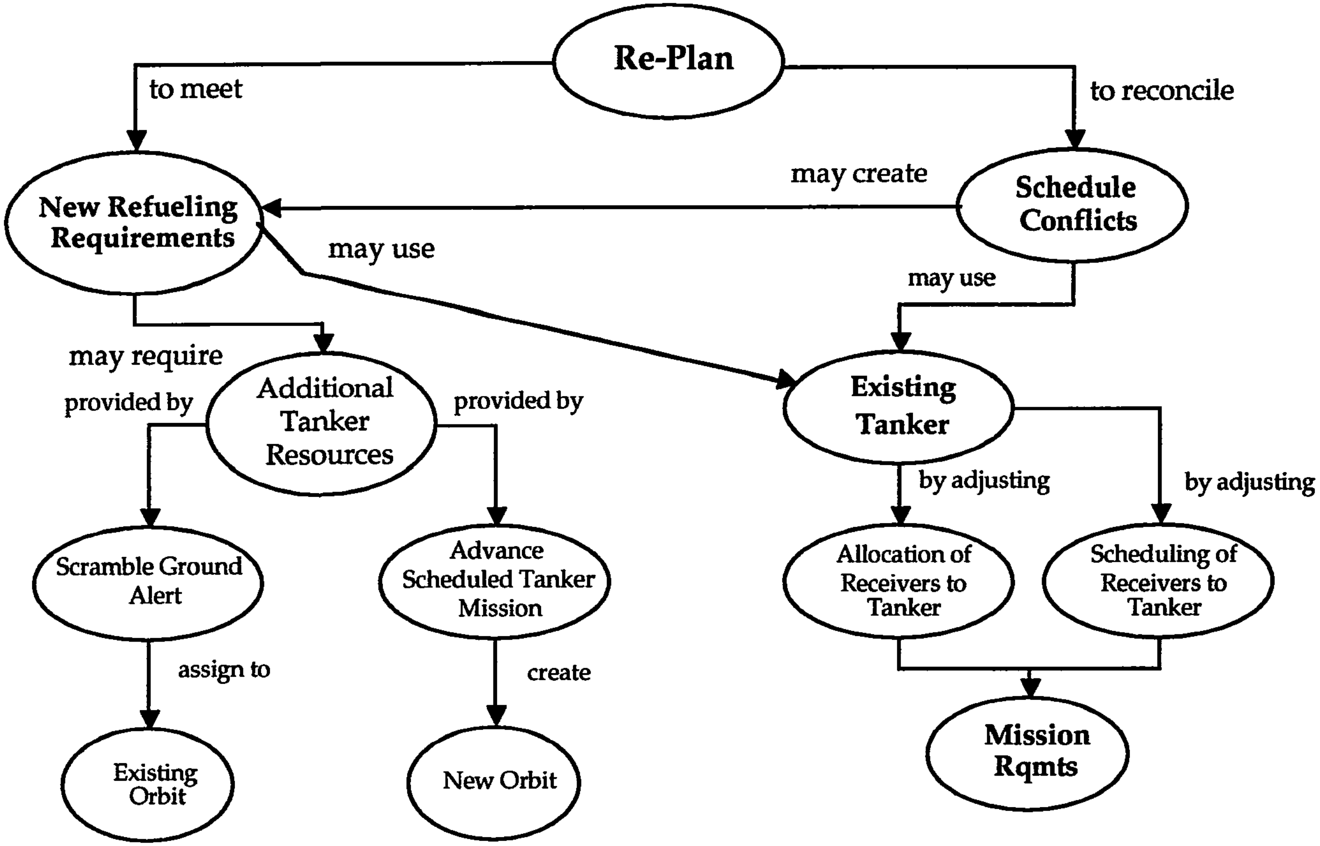


Figure C-19: Re-Planning Tasks

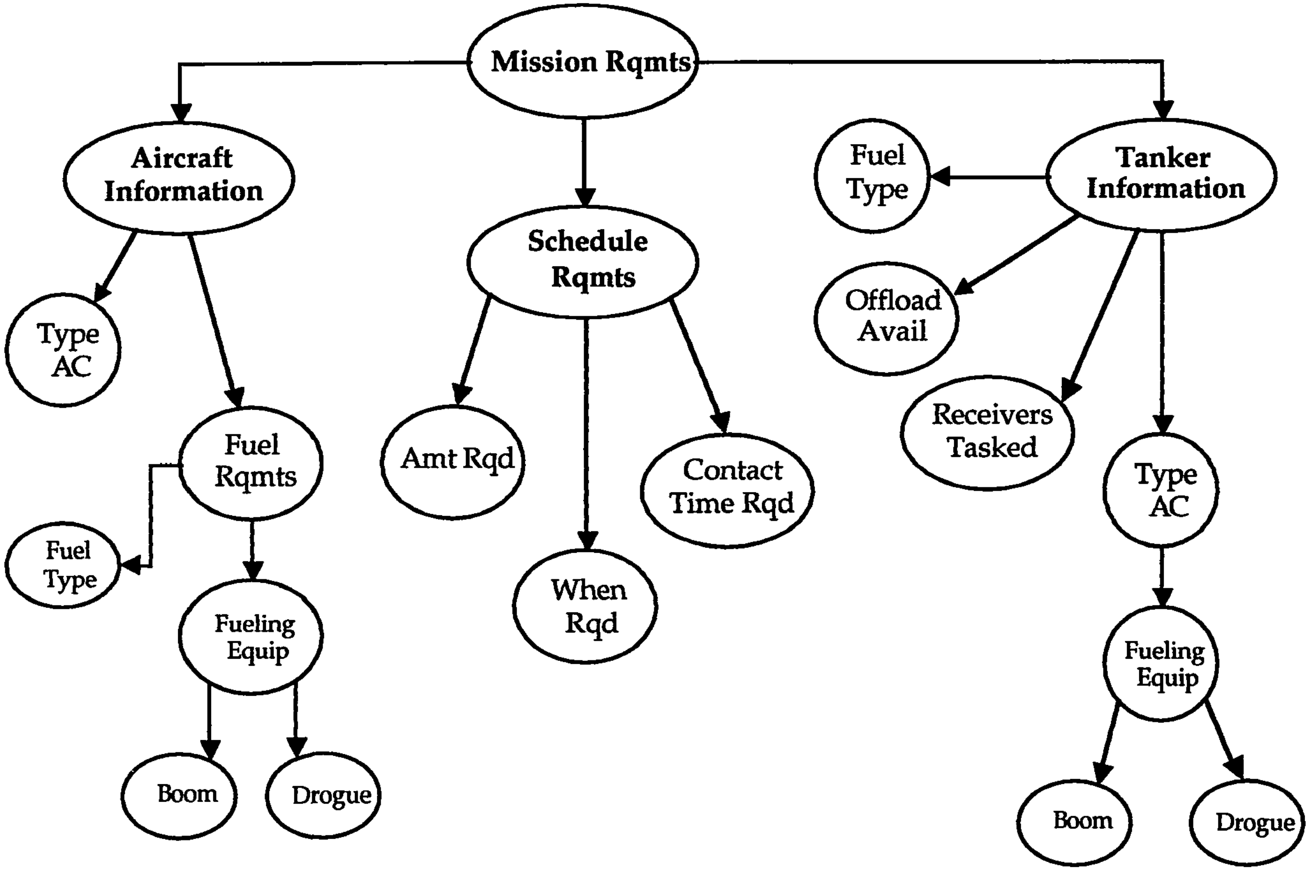


Figure C-20: Mission Requirements Information

Appendix D:

Documentary Sources Used for FLEX Case Study Requirements Identification & Modeling

Documentary Sources Used for FLEX Case Study Requirements Identification & Modeling

System Development Documents and Manuals

Advanced Planning System (APS). *System Specification for the Advanced Planning System (APS).* [PX-15666. Contract #F30602-88-0105. Prepared for Rome Laboratory, Advanced Concepts Branch, Griffiss AFB, NY.] St. Paul, MN: Unisys Corporation, 21 Feb 1991.

Advanced Planning System (APS). *Software Users Manual for the Advanced Planning System (APS) Phase V.* [PX-15978. Contract #F30602-91-C-0148. Prepared for Rome Laboratory, Advanced Concepts Branch, Griffiss AFB, NY.] St. Paul, MN: Paramax Systems Corporation, 2 Oct 1992.

Advanced Planning System (APS). *Operator Familiarization Course Training Materials for the Advanced Planning Systems (APS) Phase V.* [Contract #F30602-91-C-0148. Prepared for Rome Laboratory, Advanced Concepts Branch, Griffiss AFB, NY.] St. Paul, MN: Paramax Systems Corporation, 2 Oct 1992.

Common Mapping, Charting, Geopositioning, and Imagery System (CMS). *CMS Users Manual, Software Version 1.1.* [Final Report. LMSC-F414838; Contract # F30602-88-C-0105. Lockheed Missiles and Space Company.] Austin, TX: Lockheed Missiles & Space Company, 3 May 1991.

Computer Assisted Force Management System (CAFMS). *Computer Assisted Force Management System (CAFMS) Reference Guide, Version 6.1.* CAFMS Support Division, Directorate, Computer Systems Support, Headquarters Tactical Air Command, Langley AFB, VA. 28 Aug 1989.

Contingency Tactical Air Control System Automated Planning System (CTAPS). "CTAPS Data Flow." Hampton, VA: SAIC, 29 Jan 1993.

Contingency Tactical Air Control System Automated Planning System (CTAPS). "CTAPS: The Key to TBM Success." [Briefing by Col. Steve McCulloch, USAF, ESC/SZ], Jan 1993.

Force Level Execution (FLEX) System. *FLEX Software Design Document. Final.* [ADS-TR-09006-182-02. Contract #F30602-92-C-0054. Prepared for Rome Laboratory, Advanced Concepts Branch, Griffiss AFB, NY.] Mountain View, CA: Advanced Decision Systems, 29 Sept 1993.

Force Level Execution (FLEX) System. *FLEX System/Segment Specification.* [TR-09006-182-01. Contract #F30602-92-C-0054. Prepared for Rome Laboratory, Advanced Concepts Branch, Griffiss AFB, NY.] Mountain View, CA: Advanced Decision Systems, 10 Aug 1992.

Force Level Execution (FLEX) System. *FLEX Replanning , Prototype 2.* [Contract #F30602-92-C-0054. Prepared for Rome Laboratory, Advanced Concepts Branch, Griffiss AFB, NY.] Mountain View, CA: Advanced Decision Systems, 5 Jan 1993.

Rapid Application of Air Power (RAAP). *RAAP Software System Specification; Software Rehost (Phase IIIA). Draft.* [88-44008-007. Prepared for E-Systems Greenvile Division, Greenville, TX 75402-6056.] Plano, TX: Merit Technology, Inc., 7 Mar 1991.

Rapid Application of Air Power (RAAP). "RAAP Dynamic Profiling Concept, Vol. I." Dallas, TX: E-Systems, January 1987.

Rapid Application of Air Power (RAAP). "RAAP Phase II Plan." Dallas, TX: E-Systems, (undated).

Technical Reports

Blunden, Lt Col R. J., Jr., USAF. "Tailoring the Tactical Air Control System for Contingencies." [Research Report No. AU-ARI-91-2]. Maxwell AFB, AL: Air University Press, June 1992.

Clark, T. A., G. H. deWalder, E. C. LaBatt, G. C. Ruigrok. *FLEX ATTD Requirements Analysis Technical Report.* Griffiss AFB, NY: Rome Laboratory, Advanced Concepts Branch, 21 Feb 1992.

Clark, T. A., G. H. deWalder, E. C. LaBatt, G. C. Ruigrok. "Force Level Execution (FLEX): Developing and Demonstrating a Monitoring and Execution Capability for Combat Operations. Griffiss AFB, NY: Rome Laboratory, Advanced Concepts Branch. (undated)

Deptula, Lt Col D. A., USAF. "Planning the Air Campaign: How to Improve the Process." Memorandum for the Record, 20 Nov 1991.

Fernandes, K. *User Interface Specifications for Navy Command and Control Systems, Version 1.1.* Naval Command, Control, and Ocean Surveillance Center; Research, Development, Test and Evaluation Division, June 1992.

Linden, T. A., C. A. O'Reilly, and R. M. Tong. *Initial Combat Operations Replanning Design (ICORD) Conceptual Design.* [RL-TR-91-358. Contract # F30602-88-D-0004. Prepared for Rome Laboratory, Advanced Concepts Branch, Griffiss AFB, NY.] Bethpage, NY: Grumman Aerospace Corporation, December 1991.

Manning, H. T., M. T. Schmitt, R. A. Barker. *Future Window of the Battlefield.* [RADC-TR-89-353. Final Technical Report. Contract # F30602-87-D-0095. Prepared for Rome Laboratory, Advanced Concepts Branch, Griffiss AFB, NY.] Seattle, WA: Boeing Aerospace Company, January 1990.

Montanaro, G. D., W. J. Schroeder, B. Yamrom, W. Lorensen, D. Moitra, and P. M. Meenan. *Graphical Requirements for Force Level Planning, Volume 1 . Final Technical Report.* [RL-TR-91-239, Vol I. Contract # F30602-89-C-0078. Prepared for Rome Laboratory, Advanced Concepts Branch, Griffiss AFB, NY.] Schenectady, NY: General Electric Company, September 1991.

Moitra, Dipen, G. D. Montanaro, C. Chalek. *Graphical Requirements for Force Level Planning, Volumes 2 . Final Technical Report.* [RL-TR-91-239, Vol II. Contract # F30602-89-C-0078. Prepared for Rome Laboratory, Advanced Concepts Branch, Griffiss AFB, NY.] Schenectady, NY: General Electric Company, September 1991.

Peters, J. I, and S. P. Masterson. *C^3 Lessons Learned: Guidance for SDI C3 Center Development.* [SAIC-87/1824] McLean, VA: SAIC, October 1987.

Raphael, T. D. *Cognitive Flexibility and Innovation in the Problem Definition Process for Tactical Command and Control.* [D-39W-91 (Revised). Prepared for Logistics Research Group, Air Force Human Relations Laboratory, Wright-Patterson AFB, OH 45433-6573.] Falls Church, VA: Mystech Associates, 7 May 1991.

TRW Defense Systems Group. *Functional Description (Final) Tactical Expert Mission Planner (TEMPLAR).* [Contract # F30602-85-C-0249. Prepared for Rome Air Development Center (RADC), Griffiss AFB, NY.] March 1987.

United States Air Force. *380th Air Refueling Wing Command Post and Scheduling "As Is" Models for the Air Mobility Command (AMC).* [E-21936U. Contract #F19728-89-D-0011. Prepared for Electronic Systems Center, HQ Air Force Materiel Command, Hanscom AFB, MA.] Andover, MA: Dynamics Research Corporation, 11 Jan 1993.

Military Standards & Guidelines

Dept. of Defense. *HCI Style Guide, Version 2.0. Final Draft.* 30 Sept 1992.

Dept. of Defense. *Military Standard: Defense System Software Development. DOD-STD-2167A.* Washington, DC: February 1988.

Dept. of Defense. *Military Standard. Human Engineering Design Criteria for Military Systems, Equipment and Facilities. MIL-STD-1472D.* 14 Mar 1989.

Dept. of Defense. *Military Handbook. Human Engineering Procedures Guide. DOD-HDBK-763.* 27 Feb 1987.

Other Government Documents

Dept. of Defense. *Joint Command and Control Warfare Staff Officer Course. Student Text.* Norfolk, VA: National Defense University, Armed Forces Staff College, January 1993.

Dept. of the Air Force. *Battle Staff Course Workbook.* Air Ground Operations School, Hurlburt Field, FL, Feb 1993.

Dept. of the Air Force. *Tactical Air Operations. TAC Regulation 55-45.* Langley AFB, VA: Headquarters Tactical Air Command, 8 Apr 1988.

Dept. of the Air Force. *Tactical Air Operations. TAC Manual 2-1* Langley AFB, VA: Headquarters Tactical Air Command, 15 Apr 1978.

Appendix E:

Issues Raised in FLEX Requirements Review

Issues Raised in FLEX Requirements Review

Section 1.0 - 5.0 present a detailed review of the requirements identification process using the tables in Appendix B. Section 6.0 summarizes the design goals identified.

1.0 Environmental/Situational Profile

1.1 SITUATIONAL/ENVIRONMENTAL CONTEXT CATEGORIES (TABLE B-1)

Characteristics

- Situational context is severely stochastic -- any given situation may have a very large number of possible outcomes with roughly equivalent likelihoods.
- Adversarial component of the problem approaches the indeterminate.
- Outcomes not fully controllable by manipulating initial conditions.
- High degree of variability in all plan components.
- Non-controllable environmental conditions; non-controllable intelligent adversary.
- Result: the question is not *whether* the ATO will unravel, it is *how much, in what ways*, and *when* it will unravel.

Decision-Maker Response Strategies

- Contingency planning to handle evolving situation.
- Maintaining an ability to rapidly exploit opportunities.
- Information review for novel event classification & situation assessment.

1.2 SYSTEM DESIGN ISSUES BY CONTEXTUAL CATEGORY (TABLE B-2)

Design Goals

- Support rapidly adaptive response.

- Trade off some control for flexibility.
- Support pattern-matching, analogical reasoning, and other means for improving assessment in novel situations.
- Support for rule-based (SOP) and knowledge-based decision-making to handle situational variety.

Potential Errors

- Problems with sluggish response due to latency between event and recognition (also feedback delay).
- Adaptive strategies (rapid re-tasking, etc.) may be difficult to coordinate and control.
- Organizational learning may be impaired by lack of repeated experience.

Information Requirements

- System-level (i.e., tanker operations) displays to convey interdependencies and operational overviews.
- Ability to adaptively filter information to permit the required abstraction level, while retaining rapid access to detailed information.

1.3 DEGREE OF STRUCTURE AND BOUNDEDNESS IN DECISION CONTEXT & TASKS (TABLE B-3)

Boundedness

- Tends toward unbounded ("open").
- Most of the information load under routine conditions is tractable for the well-trained and highly motivated TDO; under combat conditions (2000 sortie ATO) the tasks become intractable -- information load exceeds human ability to absorb and manipulate.
- Decision variables are generally representative, but varying in reliability due to timeliness or inherent ambiguities.
- Situational picture becomes less reliable as multiple changes to the ATO are contemplated during combat situations.

Structure

- Semi-structured (with some aspects of unstructured problem).
- Tanker operations generally quantifiable, but certainty varies widely.

- Not all critical information is readily available -- there are both "known unknowns" and "unknown unknowns" .

1.4 LEVEL OF COMPLEXITY IN DECISION CONTEXT & TASKS (TABLE B-5)

- Complexity ranges from moderately high to very high depending upon the nature and size of the operation.

Structural Features

- Moderate vertical complexity and very high horizontal complexity; vertical complexity shifts to very high in joint and combined operations.

Interdependency

- Moderate to high dependency overall.
- Tanker operations are highly dependent; tanker operations are the "tent pole" in air operations; majority of sorties require refueling.
- Network of dependencies:
    - Inability to meet refueling requirements will result in cancellation of missions (direct dependency) with a ripple effect upon the missions which they support (indirect dependencies)
    - Airborne taskable fuel dependent on actual offloads
    - Actual offloads dependent upon aircraft and missions

2.0 Organizational Profile

2.1 ORGANIZATIONAL RESPONSE SHIFTS (TABLE B-6 A)

Routine Operations

- Situational Context is determinant to moderately stochastic; low threat, relatively static environment; longer decision horizon.
- Tend to be more formal; operations tightly controlled.
- Response to situations follows more rigid procedures based on specific guidance; TDO is less likely to exercise high degree of personal initiative.
- During non-crisis operations, TDO may be ill-prepared for sudden shift in environment to a combat state.
- Routine operations afford little opportunity to develop a range of adaptive responses -- TDO never has to push the system to the limit.

- Control is communication-dependent; communication delays between levels of hierarchy lengthen time between decision and action.

Design Guidance:

- Optimize for faster communication to minimize authorization delays under tighter control conditions.
- Make explicit all constraints/guidance (e.g., ROEs, etc.) from superiors in boundary displays, thresholds, and other conformance guide representations.
- Display structural information (i.e., functional cause & effect relationships) to aid development of mental models and support wider knowledge of response options.

Crisis Operations

- Situational Context is severely stochastic to indeterminate; high threat, highly environment; very short decision horizon.
- Operational control is loosened to facilitate rapid, adaptive response; informal structures within the COD may dominate the formal structures.
- As COD workload increases TDO will exercise more individual initiative; relaxation of control may result in local satisficing (solving the sub-unit problem at the cost of larger goals).
- TDO has opportunity to extend repertoire of response options as the system is tasked at peak levels.
- Communication delays may impair information gathering and decision implementation required for more adaptive responses.
- Intra-COD communication greatly increased; central role of tanker ops results in a barrage of task alerts to the TDO.

Design Guidance

- Optimize to provide local DM most accurate, relevant information and technological means to combine and interpret abstract/symbolic information.
- Provide doctrinal/procedural overview displays to support interpretation of and effective response to novel or rare events.

- FLEX tanker modules must help to maintain overall control to meet refueling objectives without direct review of every decision by senior command; provide organizational objectives or goal-based overview displays to prevent cognitive "tunnel vision".

3.0 Decision-Maker Profile

3.1 DECISION-MAKER'S DOMAIN KNOWLEDGE (TABLE B-7)

Characteristics

- TDO will typically have moderate to moderately-high domain understanding depending upon individual combat operations experience and completion of staff officer's course; may have wing-level but not force-level mental models.
- TDO will have situational models of domain mostly gained through instruction and exercises and should recognize most prototypical situations; TDO's without ops experience at the wing or force level will not generally possess wholistic domain models.
- TDO will generally structure goals based upon learned procedures, direct guidance, and situational models of domain and task.

Potential Errors

- TDO may misinterpret situational cues due to limitations of mental models or fixation on most available situational models.
- Limitations in domain understanding may limit TDO's ability to resolve conflicts between situational models.
- TDO may fail to recognize the degree of uncertainty in current information or the impacts of aggregated uncertainties on the viability of the plan.

Design Guidance

- Provide access to displays formatted to present situational information & operational dependencies in context of domain models.
- Support the construction of more robust mental models with option to view deeper levels of explanation.
- Make the sources and extent of domain uncertainty explicit.

3.2 DECISION-MAKER'S FUNCTIONAL TASK KNOWLEDGE (TABLE B-8)

Characteristics

- TDOs functional task knowledge will depend upon previous experience in combat ops (force and wing level) and training (schools & exercises).
- TDOs will typically exhibit high ability to perform routine procedures and moderate to moderately-high adaptability under increased workload and novel situations.

Potential Errors

- Fixation on task features that match well-known (or vividly remembered) situations may prevent DM from correctly diagnosis situation; misdiagnosis may result in the misapplication of a learned response.
- In high volume situations, TDO may not have adequate task models to filter relevant information.
- TDO may be overly confident in correctness of response due to:
  - Inadequate consideration of the network of interdependencies that make up the current situation and effect the success of the plan.
  - Failure to recognize the aggregated errors in subtasks (microdecisions) performed in multistage decisions .
  - Failure to revise plan adequately when situation changes.

Design Guidance

- Make tanker operation task constraints and affordances visible.
- Provide goal- or decision-oriented displays to focus attention on relevant information.
- Provide explicit information on the potential effects of subtask uncertainty.
- Provide option to use supports & reminders.

3.3 DECISION-MAKER'S SYSTEM INTERACTION KNOWLEDGE (TABLE B-9)

Characteristics

- TDO will most likely be a casual to competent system user.
- TDOs with less system experience may be confused by their system operation errors.

- TDOs may not know shortcuts to speed up performance of learned procedures; increased workload will result in greatly impaired performance for all but simplest tasks; competent user will be able to adapt well-understood processes to increased workload.

Potential Errors

- Casual users will forget training without use, make mistakes & slips.
- More competent users make mistakes by misapplying learned procedures.
- Users may make modal errors due to a misunderstanding about current system state.
- Users may "get lost" in the system, finding themselves in unfamiliar windows or locked out while the system performs an unintended procedure.

Design Guidance

- Provide interface features (e.g., overview screens) to help user develop mental models of system operation.
- Make system state explicit, allow users to readily determine available options.
- Minimize modal errors through constraints and consistent operations across all modes.
- Provide "undo" & "back up" capability to ensure users feel confident in exploring to extend their system knowledge; prevent "fatal" errors.
- Make use of natural or domain knowledge in the interaction symbology to allow the user to interact with the task in the most familiar terms.
- Design levels of help to permit the user to select the depth of explanation desired.
- Provide "defaults" to allow the less experienced user to access most system features with only partial knowledge.

4.0 Functional Task Profile

4.1 OUTPUT CHARACTERISTICS (TABLE B-11 A)

- TDOs discrete output unit is the response to a task request for air refueling (AR) support; in a larger sense the task output is also the overall status of the air refueling plan or the tanker operations system.

Number of Output Units

- TDO is required to respond to a high volume of AR task requests as rapidly as possible.
- TDOs will be extremely intolerant of slow system response or highly complex routines for relatively simple tasks.

Number of Elements per Unit

- Air refueling plans have multiple components (see also multiple steps below).
- TDOs need system supports to prevent their losing track of all relevant plan components; need ability to move through various levels of detail; need support for structuring components to aid in analysis.

Duration Output Unit Maintained

- Response to a specific AR task has only minimal duration; the air refueling plan should remain viable as long as possible.

Output Workload

- TDO's output workload is high during combat operations primarily due to throughput required and number of components that must be handled.

4.2 RESPONSE CHARACTERISTICS (TABLE B-11 B)

- TDO's response goals are to meet the air refueling requirements of the ATO and maintain a viable air refueling plan for as long as possible.

Goal Attainment Difficulty

- TDO's short-term and overall goals are very difficult to attain.
- System should be designed to offload the TDO of as much of the workload as possible (e.g., allocation of table look-up and computational tasks to machine).

Response Precision

- Some of the subtasks require very high precision (e.g., keeping track of taskable fuel).
- Much of the precision requirement can be allocated to the machine component; the detailed data required for response precision can be maintained and manipulated by machine.
- Automated updates relieve the TDO from being overwhelmed by the detail.

Response Frequency

- TDO response frequency during the execution of a major combat ATO is very high; AR tasks and changes to tanker operations will pile up and must be prioritized to ensure the most important are handled as rapidly as possible.
- Delays in feedback (external or internal to COD) may impair the TDO's timely response.

Simultaneity of Subtasks

- TDO must simultaneously handle the current AR tasks using FLEX while remaining a part of the off-line COD activity (e.g., incoming messages from other sources, conversations with other duty officers, etc.).
- AR tasks arrive as discrete messages, but may have to be handled by considering the planning implications of several changes simultaneously.
- TDO may have task interrupted by higher priority task.
- System must support the TDO's maintenance of situational awareness & task continuity, and complement the team activities of the COD.

4.3 PROCEDURE/SUBTASK CHARACTERISTICS (TABLE B-11C)

Number of Procedural Steps

- Handling a single AR task involves several steps, including the possibility of activating a ground alert tanker mission or creating a new tanker mission to resolve major changes to the AR plan.

Dependency of Procedural Steps

- AR subtasks are moderately dependent in terms of temporal order (either due to system or procedural constraints) and logical relationships; subtasks are highly dependent with respect to the overall AR plan.
- Overall dependency of AR plan is such that the complexity of relationships exceeds TDO ability to handle without support.
- TDO needs a way to "step back" from current situation to see the AR plan as a whole and understand the various direct and indirect dependencies.

Adherence to Procedures Required

- Certain subtasks require strict adherence to set procedures; other subtasks may be handled in so many ways that a strict procedure is not prescribed.
- System should be designed to constrain TDO from not adhering to critical procedures and make those constraints visible to the TDO; in contrast, where flexibility is allowed, the system should facilitate the TDO's ability to manipulate the options and make the affordances visible.

Procedural Complexity

- AR tasks' procedural complexity is moderately high to very high due to the number of subtasks potentially involved and the dependencies between them.

4.4 INPUT CHARACTERISTICS (TABLE B-11D)

Stimulus Variability

- Many of the input characteristics in the AR task are moderately predictable due the consistency of operational procedures, basic situational stability, etc.
- Some inputs characteristics vary widely in predictability due to inaccuracy of supporting data or novelty of situation.
- TDO may need to be reminded of the less predictable aspects of the task to ensure that proper attention has been paid to the immediate contingencies ("what-ifs").
- Variations which follow known patterns under certain conditions may be stored as templates to support faster recognition.

Stimulus Duration

- AR Task remains an open issue until changed by the TDO's response.
- TDO may need to review open requests and reorder priority under heavier workloads.

Occurrence Regularity

- AR tasks are triggered in a very irregular fashion; TDO generally cannot predict the flow of AR tasks with other than very gross metrics.

Decision-Maker's Control of Stimulus

- TDO cannot control the occurrence of the stimulus (AR task), but can control the order of response among tasks of the same priority.
- Alarms may be shut off; incoming AR tasks may be acknowledged and set aside for later response.
- As above, TDO may need to review open requests and reorder priority under heavier workloads.

4.5 FEEDBACK CHARACTERISTICS (TABLE B-11E)

Decision-Maker's Control of Response Lag

- TDO must respond to some AR tasks immediately, other tasks may be responded to within a set period of time.
- TDO needs to know when tasks will become critical to help in prioritizing numerous tasks with the same priority.

Feedback Lag

- Feedback to the TDO regarding effects on tanker operations from actions taken is delayed by as much as hours; direct feedback from other COD DOs is rapid.
- Feedback reference may be ambiguous as actions taken early in ATO day may be superseded by later events before feedback reaches TDO.
- TDO needs means to model potential effects of actions against current situation.

Reaction Time/Feedback Lag

- Reaction time for decisions is much less than feedback lag; TDO may have to make many dependent decisions long before feedback on one decision is received.
- TDO may over- or under-compensate adjustments to AR plan due to feedback lag.

Number of Choice Subtasks

- More than 50% of TDO subtasks are involve decisions based on feedback from previous responses.
- Later in the ATO day, TDO plan refinements may be entirely dependent upon the projected effects of plan changes for which there has been only partial feedback.
- TDO learning about effectiveness of their decisions may be flawed by false assumptions due to feedback lag; may generate inaccurate mental models regarding cause and effect relationships.
- TDO needs support for trying (and retracting) solutions before committing to decisions.

5.0 Decision Task Profile

5.1 STIMULUS (TABLE B-12A)

Attentional Requirements

- TDO monitors the tanker operations at random intervals to maintain situational awareness and when update alerts or task requests are received.
- Important tanker operations information may exist on multiple screens; TDO needs to have recent changes brought to his attention.

Detection Difficulty

- TDO will have no difficulty detecting discrete requests.
- TDO will have considerable difficulty detecting underlying trends in tanker operations due to variations in the timeliness of updates to key variables.

Level of Abstraction

- Current FLEX information on tanker operations exists primarily as detailed data tables; summary information available in status display board; marquee aggregates some of the operational dependencies.
- TDO needs ability to display integrates tanker-receiver dependencies, mission flows on all active tanker orbits and fuel available.
- TDO needs ability to display and compare optional configurations.

Qualitative vs. Quantitative

- Tanker operations information is primarily quantitative; qualitative information inferred through FLEX map and marquee.
- FLEX allows the TDO to tailor displays to filter, sort, and organize information.

Memory Requirements

- In combat situations updates to tanker operations data exceed human ability to absorb or manipulate within the time requirements; FLEX automates the detailed updates.
- FLEX filtering and aggregation (see above) does not adequately reduce workload due to complexity and information volume; TDO required to do mental computation and make notes to keep track of certain variables.
- TDO needs system support to reduce off-line mental computation and other memory requirements.

Reliability & Representativeness

- Tanker operations decision variables are generally understood and representative; when data is up to date, variables are reliable.
- TDO may not fully assess the impacts of situation and options based on displayable information; there are potential "unknown unknowns" in combat operations which undermine the representativeness and reliability of standard decision variables.
- TDO may misperceive situation due to incomplete or ambiguous information may result in:
  - focus on irrelevant information;
  - selection and/or fixation on incorrect explanation or solution;

  - incorrect interpretation of cues; or
  - insensitivity to missing information.
- TDO may benefit from displays of system models or goal states to aid
  - problem identification;
  - defining causal relationships;
  - identifying missing information;
  - interpreting ambiguous cues;
  - reducing over-confidence in decisions based on uncertain information.

5.2 HYPOTHESIS (TABLE B-12B)

Situation Novelty

- TDO is familiar with all the activities of tanker operations, but there is situational novelty in the ways the variables combine in combat.
- TDO may face novel situations in joint and combined operations; unpredictability of intelligent adversary may result in unfamiliar sequence of events.
- Novelty and crush of information flow may distract TDO from seeing the underlying similarity to more familiar situations.
- Routine aspects of AR replanning may be allocated to machine processes
- TDO needs goal-oriented displays of tanker operations to maintain focus on critical variables and serve as templates to aid in forming analogies to familiar situations.
- TDO needs means of viewing the consequences of actions across the ATO including the indirect effects.

Number of Possible Hypotheses

- AR situation assessment is semi-bounded with a moderate number of hypothetical possibilities to explain current AR plan status; number of hypotheses may seem greater under heavy workload situations.
- TDO needs relief from complex detail through aggregated displays and interaction with models that help to identify the differences between the current and goal states.

Decision Horizon

- TDO performs tasks in a time-critical, quasi-real time environment; has to prioritize backlog of tasks and trade off taking more time to fully analyze situation in order to process more AR tasks in a shorter period of time.
- TDO needs "at-a-glance" displays that do not require hunting or elaborate manipulation of detail to get to the relevant information quickly.
- TDO should not be burdened with off-line computation

Inferencing Required

- Most of the inferencing required for AR replanning is within set bounds, involving well-known parameters; however, the complexities of multiple receivers and their dependent missions may require a network of inferences with varying degrees of certainty.
- Multi-dimensional network of inferences is very memory-intensive; TDO must use workload reducing heuristics that may introduce bias error.
- TDO needs displays which support inferencing based on accepted operational procedures; supports for option exploration should reduce the number of inferences and relieve the workload on TDO by showing current (and projected) state to compare with immediate and longer-term consequences across the network of tanker operation dependencies.

5.3 OPTION (TABLE B-12C)

Number of Possible Options

- Number of possible options to a given AR situation are semi-bounded (limits of available resources, etc.), but sufficient in number that the TDO faced with a large number of outstanding AR tasks may feel overwhelmed by the resulting plan complexity.
- TDO needs a means of rapidly understanding the fundamental effects of hypothesized option; support for a rapid mental simulation to accept or reject the option as feasible.

Tractability

- Evaluating AR replanning options is manually intractable under high workload situations, but problem is sufficiently bounded to allow for machine support in several areas:
  - rapid recalculation of all dependent mission data to compare options;
  - mapping of restructured dependencies and highlighting conflicts.

Goal Variability

- AR goals may shift several times in a relatively short period of time requiring a re-evaluation of priorities, updates and recalculation of projected changes in AR plans.
- Most of the conflicts and effects are predictable, but the number of conflicts spawned by small event and interdependencies make manual manipulation intractable.
- TDO needs to be able to step back from detail and view AR operations in terms of higher level goals.
- Predictable goal changes may be combined into contingency scenario templates and displayed to TDO as advance notice or incorporated into a rule-based advisor.

Evaluation Difficulty

- Uncertainties and inherent complexity make outcome values for changing AR plans difficult to calculate despite the TDOs understanding of the fundamental variables.
- TDO needs facility to quickly package responses for less complex, more routine changes.
- TDO needs tools that allow rapid scoring of options against basic criteria with pre-determined or adjustable weighting.
- TDO needs displays that model or simulate the projected consequences for a given option to compare with other relatively equivalent options.

Outcome Uncertainty

- Outcome uncertainty for most AR plan components is moderate, but predictable; the broader the scope of the plan change the less certain the outcome.

- TDO choices at time *t* may leave them more or less vulnerable at time *t* + 3; the potential vulnerability to later requirements changes (i.e., contingencies) is even more uncertain and difficult to factor into the decision.
- Combined levels of uncertainty make evaluation intractable; feedback may not be timely, goals may change several times, and there is a very high penalty for making poor choices.
- Current FLEX system does not reflect the uncertainties aggregated into projected outcomes of AR plans; system ranking of options treats all quantitative data as being 100% certain -- it is possible to have two equally ranked options while being unaware of their highly disparate levels of certainty.
- TDO needs supports for understanding the degree of uncertainty inherent in a particular option.

5.4 RESPONSE (TABLE B-12D)

Planning Required

- AR plans are operational hypotheses involving multiple assumptions and inferences about current ops and the causal relationships that predict outcomes.
- AR execution in high sortie ATOs can make use of pre-planned contingencies (inactivate orbits & routes, ground alert tanker missions, alternate recovery bases, etc.) to handle many of the plan changes; extensive replanning is required when major changes are made during execution (i.e., addition of large high-priority missions; multiple failures or resource losses).
- TDO needs support for decomposing new goals into AR subtasks and means-end restructuring of AR plans to meet new requirements.

Coordination Required

- AR coordination requires coordination with other DOs in the COD, with airborne forward control units, the affected Wings and support operations; during joint and combined operations coordination involves other services and national forces.

- AR coordination is affected by the organizational shifts that occur in crisis conditions; coordination must take place within the decision horizon
- Communication requirements for coordination (i.e., management of message traffic) impose processing load on system which constrains the design options.
- Reformatting to meet messaging standards qualitatively changes information passed and may effect its interpretation at the receiving end.
- Although coordination is handled through SODO and ATO distribution chain, TDO needs support for understanding the potential coordination ramifications of options (related to interdependencies).

Execution Control Requirements

- Execution of AR plan changes is a highly dependent, multi-phased control process.
- Multiple phases increase coordination requirements and can affect feasibility of certain options due to decision horizon; increase difficulty of tracing all possible consequences of actions taken.
- Delayed feedback may be incorrectly associated with wrong phase and cause TDO to over-correct.
- TDO might benefit from a display of goals and subgoals with current execution status.

6.0 Cognitive Task Requirements Summary

REQUIREMENTS GOALS

Support for Improved Performance

- Support rapidly adaptive response.
- Provide DM most accurate, relevant information and technological means to combine and interpret information.
- Offload DM of as much of the workload as possible.
- Support pattern-matching, analogical reasoning, and other means for improving assessment in novel situations.

Support for Distributed Decision-Making

- System must support the TDO's maintenance of situational awareness & task continuity, and complement the team activities of the COD.
- Provide means to maintain overall control to meet mission objectives without direct review of every micro-decision by senior command
- Optimize for fast communication to improve coordination and minimize authorization delays.

Support for Development of Decision-Making Knowledge

- Make use of natural or domain knowledge in the interaction symbology to allow the user to interact with the task in the most familiar terms.
- Display structural information (i.e., functional cause & effect relationships) to aid development of mental models and support wider knowledge of response options.
- Provide doctrinal/procedural overview displays to support interpretation of and effective response to novel or rare events.
- Provide varying levels of explanation to support the construction of more robust mental models.

SPECIFIC COGNITIVE TASK REQUIREMENTS

Support for Situational Awareness & Understanding

- Provide display features (e.g., overview screens) to help user develop mental models of operational environment.
- Make the sources and extent of uncertainty explicit.
- Provide templates of various known patterns and causal conditions to support faster recognition.

Support for Focus on Goal/Decision-Relevant Information

- Provide goal- or decision-oriented displays to focus attention on relevant information and support
    - identifying situation and/or problem;
    - defining causal relationships;
    - identifying missing information;
    - interpreting ambiguous cues;

  - reducing over-confidence in decisions based on uncertain information.
- Provide predictable goal changes in contingency scenario template displays.

Support for Understanding of Operational & Domain Dependencies

- Provide system-level (i.e., tanker operations) displays to convey interdependencies and situational overviews.
- Example: TDO needs ability to display integrated tanker-receiver dependencies, mission flows on all active tanker orbits and fuel available.)

Support for Reducing Mental Workload

- Provide system support to reduce off-line mental computation and other memory requirements.
- Provide option to use supports (e.g., table look-up tasks) & reminders.
- Provide and propagate automated updates relieve the DM from being overwhelmed by maintaining detail.

Support for Viewpoint Adjustment

- Provide the DM the ability to adaptively filter information to permit the required abstraction level, while retaining rapid access to detailed information.
- Provide ability to "step back" from detail and view AR operations in terms of higher level goals and the various direct and indirect dependencies.
- Provide "at-a-glance" displays that do not require hunting or elaborate manipulation of detail to get to the relevant information quickly.

Support for Option Comparisons

- Provide means of viewing the consequences of actions across the ATO including the indirect effects.
- Provide support for trying (and retracting) solutions before committing to decisions.
- Provide a means for a rapid mental simulation to accept or reject the option as feasible.
- Provide displays which support inferencing based on accepted operational procedures.

- Provide support for rapid scoring of options against basic criteria with pre-determined or adjustable weighting.
- Provide displays that model or simulate the projected consequences for a given option to compare with other relatively equivalent options.
- Provide support for understanding the degree of uncertainty inherent in a particular option.

Support for Decision Control & Guidance

- Provide means to make explicit all constraints/guidance (e.g., ROEs, etc.) from superiors in boundary displays, thresholds, and other conformance guide representations.
- Provide means to display operational goals and subgoals with current execution status.
- Provide support for decomposing new goals into subtasks and means-end restructuring of plans to meet new requirements.
- Provide facility to quickly package responses for less complex, more routine changes.
- Provide facility to quickly bring recent changes to DM's attention.
- Provide means to make task constraints and affordances visible.
- Provide facility to remind DM of the less predictable aspects of the task to ensure that proper attention has been paid to the immediate contingencies ("what-ifs").
- Provide means to shut off alarms; provide means to acknowledge tasks received and set aside for later response.
- Provide facility to review open requests and reorder priority under heavier workloads.
- Provide facility to alert DM when tasks will become critical to help in prioritizing numerous tasks with the same priority.

Support for Interface Operation & Error Control

- Provide means to make system state explicit and permit users to readily determine available options.
- Provide constraints and consistent operations across all modes to minimize modal errors.

- Provide support for rapid scoring of options against basic criteria with pre-determined or adjustable weighting.
- Provide displays that model or simulate the projected consequences for a given option to compare with other relatively equivalent options.
- Provide support for understanding the degree of uncertainty inherent in a particular option.

Support for Decision Control & Guidance

- Provide means to make explicit all constraints/guidance (e.g., ROEs, etc.) from superiors in boundary displays, thresholds, and other conformance guide representations.
- Provide means to display operational goals and subgoals with current execution status.
- Provide support for decomposing new goals into subtasks and means-end restructuring of plans to meet new requirements.
- Provide facility to quickly package responses for less complex, more routine changes.
- Provide facility to quickly bring recent changes to DM's attention.
- Provide means to make task constraints and affordances visible.
- Provide facility to remind DM of the less predictable aspects of the task to ensure that proper attention has been paid to the immediate contingencies ("what-ifs").
- Provide means to shut off alarms; provide means to acknowledge tasks received and set aside for later response.
- Provide facility to review open requests and reorder priority under heavier workloads.
- Provide facility to alert DM when tasks will become critical to help in prioritizing numerous tasks with the same priority.

Support for Interface Operation & Error Control

- Provide means to make system state explicit and permit users to readily determine available options.
- Provide constraints and consistent operations across all modes to minimize modal errors.

- Provide "undo" & "back up" capability to ensure users feel confident in exploring to extend their system knowledge; prevent "fatal" errors.
- Provide levels of help to permit the user to select the depth of explanation desired.
- Provide "defaults" to allow the less experienced user to access most system features with only partial knowledge.

Appendix F:

Examples from an Integrated System/Segment Specification (SSS) The FLEX Case Study

Examples from an Integrated System/Segment Specification (SSS): The FLEX Case Study

A subset of FLEX requirements (3.2.1.2.5 - 3.2.1.2.31) from the FLEX System/Segment Specification, August 10, 1992 (ADS-TR-09006-182-01). Requirements identified in the CSE requirements identification have been added in the format indicated below. Certain aspects have been condensed to maintain the focus on cognitive task support for air refueling operations. Although requirements are generally only added to address the tanker operations case study, all added requirements are expressed in generic terms.

Cognitive Task Requirements added by the author appear in boxes for highlighting purposes only.

Author's comments appear in italics.

3.2.1.2.5 Pre-planned ATO Execution

PURPOSE

This system capability provides the facilities which enable the operator to review, monitor, coordinate, and control activities related to the execution of the pre-planned missions published in the ATO.

REQUIREMENTS

. . .

e. Facilities shall be provided which permit the operator to monitor information and control the execution of the planned aerial refueling operations as scheduled in the ATO or its updates.

1. Display planned missions and missions supported information in an order selected by the operator.

1a. Provide means for operator to create, store, and quickly call up customized "defaults", including:

- (1.) current airborne missions and supported missions
- (2.) missions affected by change/update.
- (3.) missions in conflict (current or projected conflict)
- (4.) missions in particular area (e.g., battlefield quadrant, target region)
- (5.) missions by service or nationality
- (6.) missions requiring re-planning or adjustment in order of urgency
- (7.) untasked airborne or ground alert missions available (provide ability to select by region as in 4 above)

2. View base/unit sorties flows, asset availability, and track flow schedules in a graphical form.

2a. Provide means for operator to comprehend key mission information quickly , including:

- (1.) means for operator to orient quickly on mission sortie flow displays with respect to current ATO time.
- (2.) means for operator to quickly determine currently airborne missions.
- (3.) means for operator to quickly determine day/night status.
- (4.) means for operator to quickly recognize the content and impact of updated information.

2b. Provide means for operator to view entire ATO day with lower detail.

3. View base/unit sortie flows, asset availability, and track flow schedules in a table form.

a. Provide means for operator to create, store, and quickly call up customized "defaults" (see 1a above).

4. Display refueling locations/tracks as they relate to threats on a map graphic.
5. Review unfulfilled offloads and tasking.
    a. Provide means for operator to create, store, and quickly call up customized "defaults" (see 1a above).
6. Display refueling locations/tracks as they relate to asset protection on a map graphic.

3.2.1.2.6 Execution of Immediate Tactical Air Missions

PURPOSE

This system capability provides the facilities which enable the combat operator to review, monitor, coordinate and control activities related to the execution of immediate air operations.

REQUIREMENTS

. . .

b. Facilities shall be provided which permit the operator to monitor, coordinate and execute the air resources available for immediate tactical air operations.
c. Facilities shall be provided which permit the operator to monitor, coordinate and execute the air resources to conduct near future air operations.

. . .

f. Facilities shall be provided which permit the operator to monitor and coordinate for required tactical support of immediate airlift movements.
g. Facilities shall be provided which permit the operator to review, monitor, and coordinate inflight reports.

3.2.1.2.7 Monitor Friendly Forces

PURPOSE

This system capability provides facilities which enable the operator to review and monitor the friendly forces situation.

REQUIREMENTS

a. The system shall provide facilities which permit the operator to review and monitor current and cumulative data on the following areas:
    1. base/unit aircraft operational status
    2. base/unit weapons availability status by type and SCL
    3. unit aircrew status
    4. airfield operational status
    5. base/unit mission flying schedule
    6. base/operating location current weather
    7. air defense warning status

3.2.1.2.8 Monitor Enemy Forces

PURPOSE

This system capability provides the facilities which enable the operator to review and monitor the enemy forces situation.

3.2.1.2.9 Monitor Current Operations

PURPOSE

This system capability provides the facilities which enable the operator to review, monitor, and coordinate the status for units under the control of the ACC and which appear in the ATO.

REQUIREMENTS

a. Facilities shall be provided which permit the operator to review, monitor, and coordinate the following areas:
    1. View alert schedules and monitor the course of their execution.
    2. Display projected mission takeoff times as updated by the unit.
    3. Monitor desired aircraft/weapon system current operational status
    4. Review existing mission reports.
    5. Monitor current and trend logistics status of execution and support weapons systems.
    6. Monitor current air defense warning condition.

3.2.1.2.10 Analyze Friendly Situation

PURPOSE

This system capability provides the facilities which enable the operator to review and monitor data which will facilitate the analysis of the impact of changes in friendly resources or environment on the capability to execute the planned ATO.

REQUIREMENTS

a. The system shall provide facilities which permit the operator to detect changes in available operational resources which exceed a previously defined threshold.

> The system shall provide facilities which permit the operator to detect the content and impact of changes in available operational resources which exceed a previously defined threshold.

b. The system shall provide facilities which permit the operator to display changes in available operational resources which exceed a previously defined threshold.

> 1. The system shall provide facilities which permit the operator to rapidly display changes in available operational resources with operator-defined filters.

c. Facilities shall be provided which permit the operator to detect when actual resource changes to data used to plan the ATO occur.

> The system shall provide facilities which alert the operator when changes to data affect resources planned in the ATO.
>
> 1. Alerts shall inform the operator of the urgency of the change.
> 2. The system shall provide facilities to permit the operator to acknowledge the alert depending upon urgency, including
>
> (a.) acknowledge and continue current task
> (b.) acknowledge and display details

d. Facilities shall be provided which permit the operator to evaluate the operational capability based on new data.

1. The system shall provide facilities which permit the operator to review the impact of changes upon operational capability, including:
    - (a.) ripple effect across supported missions
    - (b.) conflicts generated
    - (c.) resources available for re-tasking
    - (d.) unfulfilled tasks
    - (e.) urgency with respect to current or near future operations

. . .

3.2.1.2.11 Analyze Current Operations

PURPOSE

This system capability provides the facilities which enable the operator to review and monitor the data which will facilitate the analysis of the effectiveness of executed operations.

3.2.1.2.12 Action Requests

PURPOSE

This system capability provides facilities which enable the operator(s) to record and manage deviations, potential deviations, or other issues that need to be brought to the attention of the Combat Operations staff. This capability will notify the appropriate Combat Operations personnel that they need to take some action in response to a deviation, and then keep track of all subsequent actions related to the deviation. This capability will help prevent important information and actions from inadvertently being omitted.

REQUIREMENTS

a. The system shall provide facilities which present the operator feedback on the status of created tasks, including
    1. receipt acknowledgment from tasked DO(s)
    2. action taken
    3. task completed

b. The system shall provide facilities which allow the operator to respond quickly to simple tasks or simply acknowledge those which do not require any further action.
c. The system shall provide facilities identify the timing constraints on the response.

3.2.1.2.13 Constraint Checking

PURPOSE

This system capability provides the facilities which enable the operator to determine the need for an adjustment in the current executing ATO based on any changes that have occurred since the ATO was developed or updated.

REQUIREMENTS

a. The system shall provide facilities which notify the operator of all missions affected by changes or updates in the ATO, including
    1. supported missions
    2. supporting missions
    3. dependent missions not affected by change
b. The system shall provide facilities which allow the operator to display missions affected by changes based upon prioritization, such as:
    1. priority of mission
    2. priority of supporting/supported missions

3.2.1.2.14 Map Data Display

Facilities shall be provided which enable an operator to display a tactical map of the are of interest. The following subsections identify the specific capabilities associated with the tactical map displays. This system capability will be reused from APS.

. . .

3.2.1.2.14.2 Situation Displays

PURPOSE

This system capability provides facilities which enable the display of digital maps representing the tactical area of interest.

3.2.1.2.14.3 Feature Overlays

PURPOSE

This system capability provides facilities to overlay tactical information on the map display.

REQUIREMENTS

a. The system shall provide facilities to overlay the tactical map with graphical information to assist the Duty Officer.

. . .

3.2.1.2.14.4 Feature Visibility

PURPOSE

This system capability provides facilities which enable the operator to control the visibility of all feature overlays (i.e., to enable or disable display of feature data).

a. The operator shall be able to select the visibility of ...

b. The operator shall be able to create, store and select preferred feature visibility defaults to filter or highlight missions/features, including:

   1. Specific ATO time range (current or near future operations)
   2. Missions/features affected by change/update.
   3. Missions/features in conflict (current or projected conflict)

. . .

3.2.1.2.14.7 Additional Graphics Functionality

PURPOSE

This section outlines additional mapping functionality.

[Note: none of these features were implemented in Prototype 3.]

REQUIREMENTS

a. The location and number of alert aircraft shall be displayed on the map.
b. Status changes need to be represented on the map (e.g., base flashing when its runway is disabled).
c. A capability to click and move icons.
d. A capability to provide some form of animation to the ABP, to include a snapshot of the current planned or actual ABP displayed on the map and the animation of several snapshots over time.

3.2.1.2.15 Status Display Boards

PURPOSE

This system capability provides facilities which enable the operator to monitor the status of the ABP and resources using automated models of current display boards.

REQUIREMENTS

a. As a starting point, the status display boards defined in TACR 55-45 shall be considered.
b. In addition, the Status Display Boards (SDB) used in CAFMS shall also be considered.
c. If more information is needed than is displayed for a mission (e.g., detailed information on the target or a cross reference to the mission's package), the operator shall have access to that information by simply "clicking" on the mission.
   [Note: implemented as a multi-step query function.]

. . .

f. The status display boards shall be automatically updated from status updates and re-planning changes.
g. Methods such as colors, highlights, flashing, etc. shall be used to signify changes in status, conflicts, problems, etc.

. . .

i. The operator shall be able to quickly display information of interest through the use of customized and stored filters, including:
   1. current airborne missions and supported missions, color-coded
   2. missions affected by change/update.
   3. missions in conflict (current or projected conflict)
   4. missions in particular area (e.g., battlefield quadrant, target region)
   5. missions by service or nationality
   6. missions requiring re-planning or adjustment in order of urgency

3.2.1.2.16 Graphical Mission Schedules (Marquee)

PURPOSE

This system capability provides facilities which enable the operator to graphically monitor the status of the ABP, enhancing or complementing the Mission Schedule status display boards.

REQUIREMENTS

a. As a starting point, the Mission Schedule status display boards defined in TACR 55-45 shall be considered for graphical display. These include:
   . . .
   4. Tanker Mission Schedule
b. If more information than is displayed for a mission is needed (e.g., detailed information on the target or a cross reference to a mission's package), the operator shall have access to that information by simply "clicking" on the mission.

. . .

e. The graphical *(marquee)* shall be automatically updated from status updates and re-planning changes.

f. Methods such as colors, highlights, flashing, etc. shall be used to signify changes in status, conflicts, problems, etc.
g. The operator shall be able to use several of these graphical displays *(marquees)* at one time.
h. The operator shall be able to quickly display information of interest through the use of customized and stored filters, including:
    1. current airborne missions and supported missions, color-coded
    2. missions affected by change/update.
    3. missions in conflict (current or projected conflict)
    4. missions in particular area (e.g., battlefield quadrant, target region)
    5. missions by service or nationality
    6. missions requiring re-planning or adjustment in order of urgency
i. The operator shall be able to compress or expand the timeline to view ATO information at desired level of detail.

3.2.1.2.17 User Alerts

PURPOSE

This system capability provides facilities which alert the operator to changes in the status (deviations) of the ABP or resources, or action that must be taken. Deadlines are used to set the time for when alerts are to be generated.

REQUIREMENTS

a. The alert cannot permanently disrupt the user's ongoing activity.
b. The alert can be both audible and visual.
c. The following types of alerts should be considered.
    1. Weather changes
    2. Re-planning changes
    3. Downed aircraft
    4. Change in Commander's Guidance
    5. Change in base/runway status
    6. Mission delays
    7. High priority immediate target requests.

d. The user will be alerted that a deadline has been reached.

e. The alert should have a readily detectable urgency classification.

. . .

3.2.1.2.23 Request & Direct Adjustments for the ATO

PURPOSE

This system capability provides the facilities which enable the operator to request, coordinate, and direct adjustments of the ATO.

3.2.1.2.24 Coordination

PURPOSE

This system capability provides facilities which enable Duty Officers to coordinate an ABP change with the SODO/CCO or others.

3.2.1.2.25 Resource Accounting

PURPOSE

This system capability provides facilities for maintaining an accurate account of all resource data that could be used for re-planning of the ABP.

REQUIREMENTS

a. The system shall provide facilities for maintaining correct status of the following:
   1. Mission requests satisfied and remaining.
   2. Number of sorties tasked versus availability at each unit/base.
   3. Fuel allocated versus remaining for each tanker mission.
   4. Unit missions tasked as a function of time.
   5. Munitions used versus remaining for each unit/base.
   6. Missions expected to overlap ATO periods.
   7. Mission results, as they affect resources.

. . .

d. Under these facilities, the system shall notify the operator when a particular resource has been fully allocated, at which time the operator will be able to override the accounting system and allow over-tasking.

3.2.1.2.26 Task Alert Assets

PURPOSE

This system capability provides facilities which enable the operator to task alert assets previously scheduled to be on ground or airborne alert. These assets are scheduled in the ATO, but not specifically tasked until the execution phase begins..

REQUIREMENTS

. . .

f. The system shall provide easy access to the following information while using this capability: graphical display *(marquee)*, status display boards, Target Nomination List (TNL), available resources, and task alert assets work area.

3.2.1.2.27 Resource Retraction

PURPOSE

This system capability provides facilities which enable the operator to free up previously tasked/scheduled resources in preparation for redirection of assets. A retraction would be required every time missions were diverted, canceled, or aborted. This function could be used when limited assets are available to handle a given situation.

REQUIREMENTS

a. Support the cancellation of scheduled missions, and account for freed-up resources by updating the correct data structures.
b. Support mission aborts, maintaining the following information:
    1. The new status of the resource (e.g., in-transit, returning to base).
    2. The approximate time the resource shall be available again (e.g., turn times).

3. Recovery point of resource.
4. Ripple effects that may occur due to the abort.

Including:
(a.) Supported missions
(b.) Supporting missions (increase in taskable resources)

c. Support mission diverts, ensuring that all new information is updated in the ABP.

Include ripple effects to the following:
1. Supported missions
2. Supporting missions (increase in taskable resources)

d. Generate action requests to responsible Duty Officer positions to handle the effects of a diversion, cancellation, or abort.

3.2.1.2.27 Resource Assignment

PURPOSE

This system capability provides facilities which enable the operator to task resources in a similar manner in which missions are refined using the planning capabilities of APS.

REQUIREMENTS

. . .

c. The following manual assignment capabilities shall be provided for attach missions:

   . . .

   4. The specification of refueling requirements for the mission.

   . . .

d. The following manual assignment capabilities shall be provided for tanker missions:
   1. The creation of tanker missions.
   2. The assignment of resources (number and type of aircraft) from a base and unit to the mission.
   3. The scheduling of the mission in time.
   4. The specification of refueling requirements for the mission.

5. The assignment of receiver missions to tanker missions.
6. The scheduling of receiver missions for tanker missions.
7. The designation of fuel offloads for receivers.

. . .

3.2.1.2.29 Force Packaging

PURPOSE

This system capability provides facilities which enable the operator to designate missions which have mutual dependencies and to group them to produce force packages.

3.2.1.2.30 ABP Retasking

PURPOSE

This system capability provides facilities to assist the operator during the tasking or retasking of ABP assets.

REQUIREMENTS

a. Assist the operator in determining the ramifications (scope) of retasking air assets.
b. Provide easy-to-use methods of accomplishing the retasking, to include:
   1. Designation of possible alert assets available for scramble or retasking.
   2. Designation of possible assets already assigned to a mission as a candidate for retasking.
   3. Retraction of resources from designated missions.
   4. Reassignment of available assets to new missions.
   5 Constraint checking on newly formed missions to ensure consistency (times, coordination points, fuel, distance, etc.).
   6. The modification of any data field contained in a mission line.
   7. Easy-to-use textual methods of modifying mission lines, including changes made on the status display boards.
   8. Easy access to the following information while making ABP modifications: graphical display, status display boards, TNL, mission

line work area, available resources, configuration options, and weaponeering options.

9. Mechanisms to control the re-planning process for complicated adjustments, such as partitioning/aggregating resources and targets.
10. Mechanisms to limit the rippling effect, or a complete "unraveling" of the original plan (e.g., only allow the retraction of resources to n levels).
11. Mechanisms to assist the backfilling of missions after resources are retracted.
12. Allow graphical modifications to the mission (point, click, drag symbols), using the graphical display. Corresponding textual displays will ensure graphical modifications are accurate.

c. The system shall maintain the state of the ABP immediately prior to re-planning and allow the operator to return the state of the ABP to what it was prior to re-planning.

d. The system shall provide easy access to the following information while accomplishing ABP retasking: graphical display, status display boards, TNL, mission line work area, available resources, configuration options, and weaponeering options.

e. Allow for the coordination and dissemination of all ABP retaskings using the Tasking Message capability.

f. Generate action requests, as necessary, to other Duty Officers for action and coordination.

g. The system shall provide the means for accomplishing simple retasking (retraction & reassignment) of assets with a minimal number of steps.

h. The system shall provide the means for advising affected DOs (i.e., "heads-up") when major retasking is being developed.

3.2.1.2.31 Options Generation

PURPOSE

This system capability provides facilities which automatically construct options which represent potential changes to the tasking contained in the ABP. The operator will have control in initiating the options by applying necessary constraints, analyzing options, and modifying the options once generated. This capability is the backbone of the automated re-planning and mix-initiative capability.

Note: Options Generation was only in the earliest stages for Prototype 3. Many of the issues addressed under Options Generation relate to the DM's requirements for evaluation and selection of options regardless of whether the options are generated manually or by the system.

REQUIREMENTS

a. Provide the operator with options in response to discrepancies or deviations.

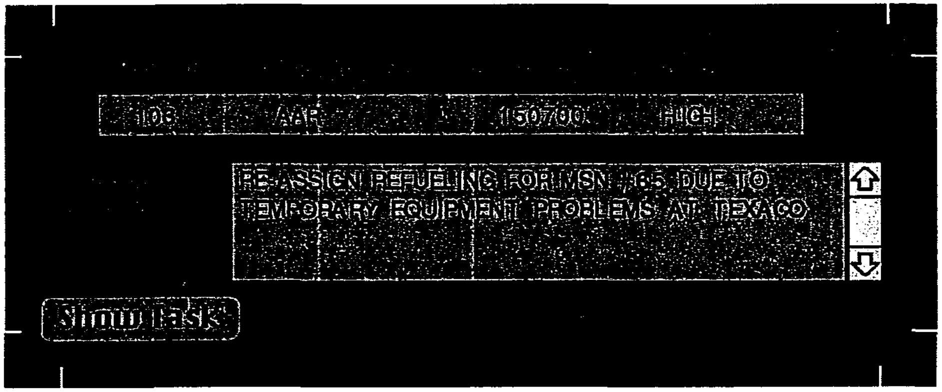
b. The options generation capability can be invoked at any time throughout the re-planning process. The receipt of an action request may be sufficient to invoke this capability.

c. The options generated shall be presented in a manner that allows the operator to make an educated decision. For example:

   1. rank of the options
   2. evaluation results
   3. any restrictions/constraints
   4. graphical representation of the effects on dependent missions
   5. ability to compare two or more options simultaneously
   6. presentation of all key decision variables in one location (one window)

Appendix G:

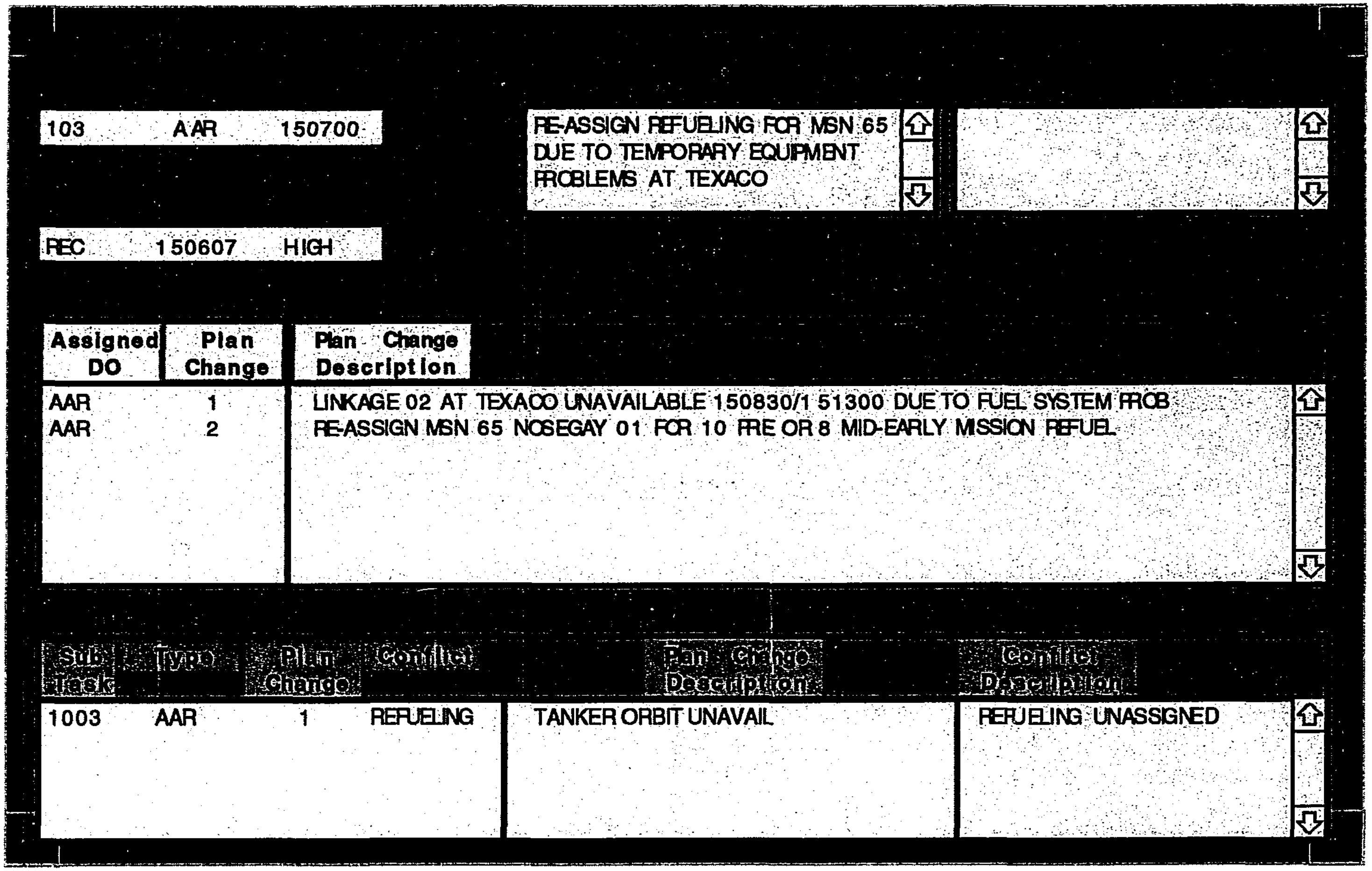
Screens from the FLEX Tanker Module Case Study



103 AAR 150700 HIGH

RE-ASSIGN REFUELING FOR MSN #65 DUE TO
TEMPORARY EQUIPMENT PROBLEMS AT TEXACO

Figure G-1: Task Notify
FLEX Window (Modified for Case Study)



103 AAR 150700

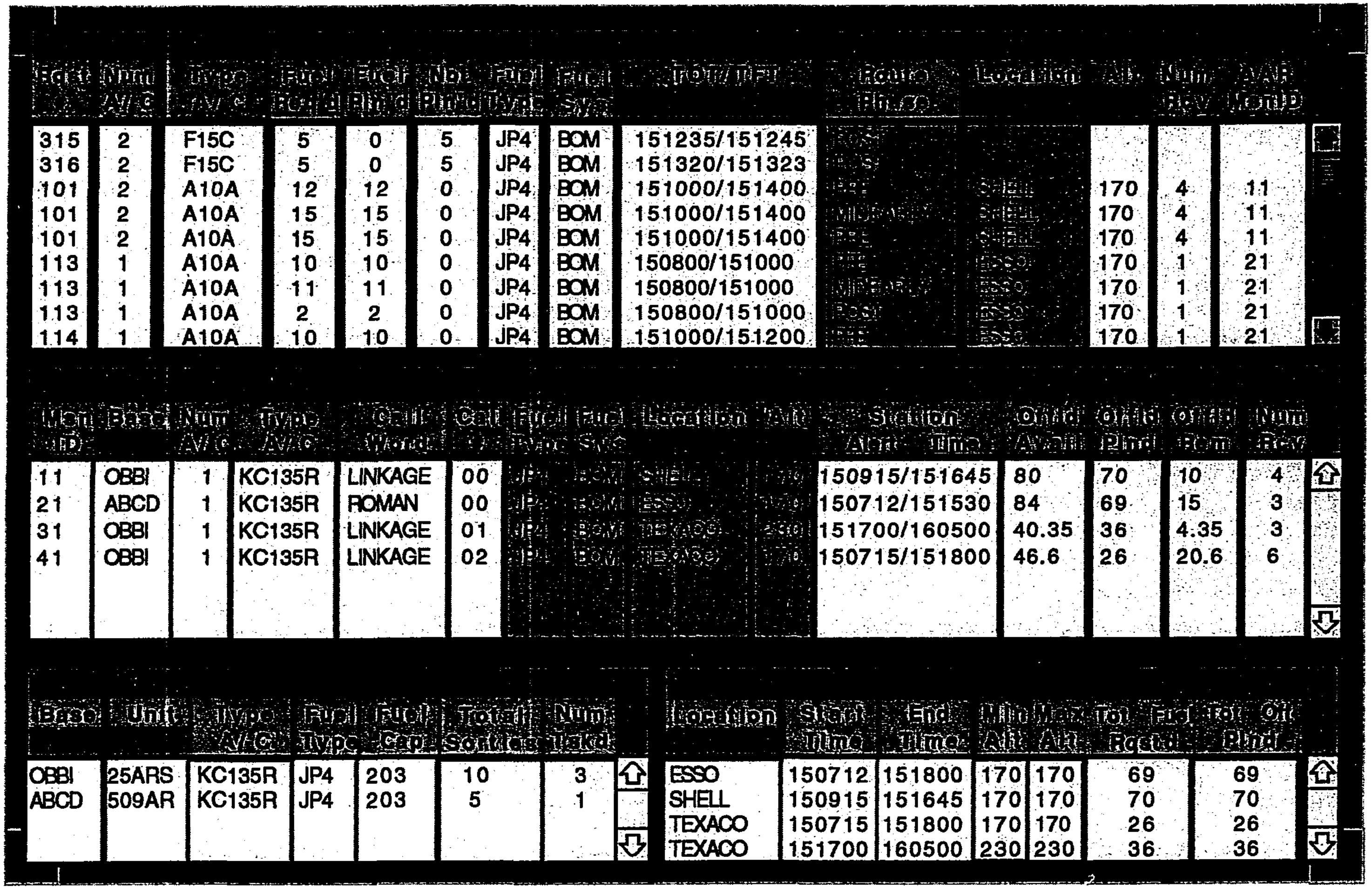
REC 150607 HIGH

RE-ASSIGN REFUELING FOR MSN 65 DUE TO TEMPORARY EQUIPMENT PROBLEMS AT TEXACO

Assigned DO	Plan Change	Plan Change Description
AAR	1	LINKAGE 02 AT TEXACO UNAVAILABLE 150830/151300 DUE TO FUEL SYSTEM PROB
AAR	2	RE-ASSIGN MSN 65 NOSEGAY 01 FOR 10 PRE OR 8 MID-EARLY MISSION REFUEL

Sub Task	Type	Plan Change	Conflict	Plan Change Description	Conflict Description
1003	AAR	1	REFUELING	TANKER ORBIT UNAVAIL	REFUELING UNASSIGNED

Figure G-2: Task Inspector
FLEX Window (Modified for Case Study)



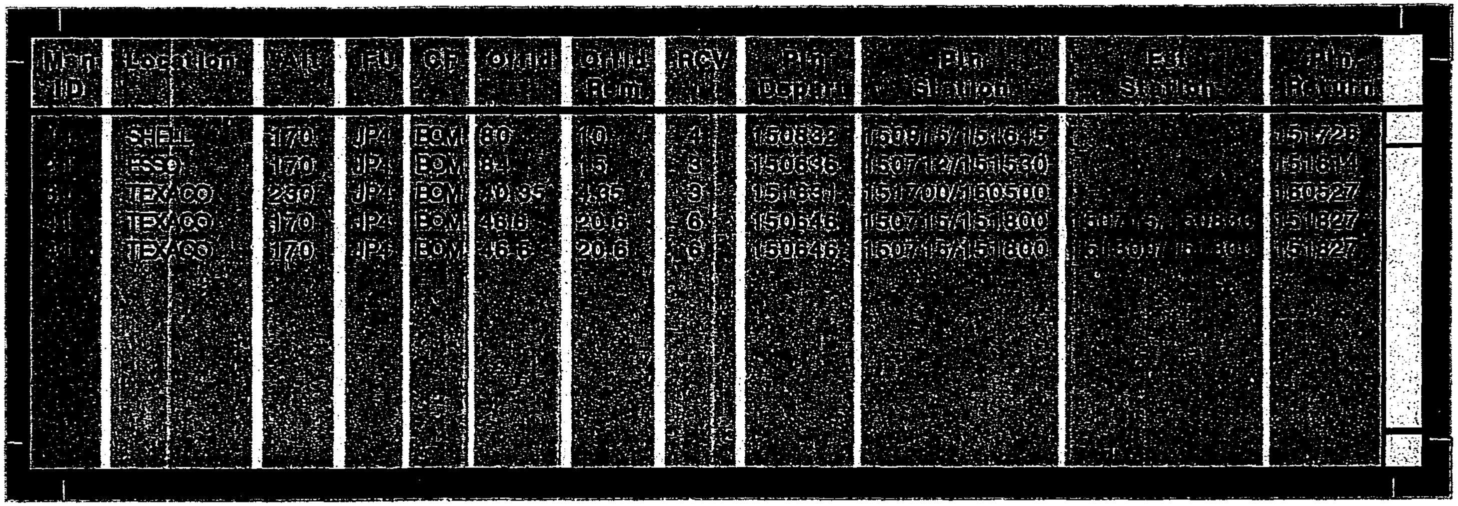
Rqst ID	Num A/C	Type A/C	Fuel Rqstd	Fuel Plnd	Not Plnd	Fuel Type	Fuel Sys	TOT/TFT	Route Phase	Location	Alt	Num Rcv	A/R Mon ID
315	2	F15C	5	0	5	JP4	BOM	151235/151245	POST				
316	2	F15C	5	0	5	JP4	BOM	151320/151323	POST				
101	2	A10A	12	12	0	JP4	BOM	151000/151400	PRE	SHELL	170	4	11
101	2	A10A	15	15	0	JP4	BOM	151000/151400	[illegible]	SHELL	170	4	11
101	2	A10A	15	15	0	JP4	BOM	151000/151400	PRE	SHELL	170	4	11
113	1	A10A	10	10	0	JP4	BOM	150800/151000	PRE	ESSO	170	1	21
113	1	A10A	11	11	0	JP4	BOM	150800/151000	[illegible]	ESSO	170	1	21
113	1	A10A	2	2	0	JP4	BOM	150800/151000	POST	ESSO	170	1	21
114	1	A10A	10	10	0	JP4	BOM	151000/151200	PRE	ESSO	170	1	21

Msn ID	Base	Num A/C	Type A/C	Call Word	Call #	Fuel Type	Fuel Sys	Location	Alt	Station Arrv/Time	Offld Avail	Offld Plnd	Offld Rem	Num Rcv
11	OBBI	1	KC135R	LINKAGE	00	JP4	BOM	SHELL	170	150915/151645	80	70	10	4
21	ABCD	1	KC135R	ROMAN	00	JP4	BOM	ESSO	170	150712/151530	84	69	15	3
31	OBBI	1	KC135R	LINKAGE	01	JP4	BOM	TEXACO	230	151700/160500	40.35	36	4.35	3
41	OBBI	1	KC135R	LINKAGE	02	JP4	BOM	TEXACO	170	150715/151800	46.6	26	20.6	6

Base	Unit	Type A/C	Fuel Type	Fuel Cap	Total Sorties	Num Takd
OBBI	25ARS	KC135R	JP4	203	10	3
ABCD	509AR	KC135R	JP4	203	5	1

Location	Start Time	End Time	Min Alt	Max Alt	Tot Fuel Rqstd	Tot Off Plnd
ESSO	150712	151800	170	170	69	69
SHELL	150915	151645	170	170	70	70
TEXACO	150715	151800	170	170	26	26
TEXACO	151700	160500	230	230	36	36

Figure G-3: Tanker Worksheet
FLEX Window (Modified for Case Study)



Msn ID	Location	Alt	FU	CF	Offld	Offld Rem	RCV	Pln Depart	Pln Station	Est Station	Pln Return
[illegible]	SHELL	170	JP4	BOM	80	10	4	150832	150915/151645		151728
[illegible]	ESSO	170	JP4	BOM	84	15	3	150636	150712/151530		151614
[illegible]	TEXACO	230	JP4	BOM	40.35	4.35	3	151631	151700/160500		160527
[illegible]	TEXACO	170	JP4	BOM	46.6	20.6	6	150646	150716/151800	[illegible]	151827
[illegible]	TEXACO	170	JP4	BOM	46.6	20.6	6	150646	150716/151800	[illegible]	151827

Figure G-4: Tanker Status Display Board (SDB)
FLEX Window (Modified for Case Study)

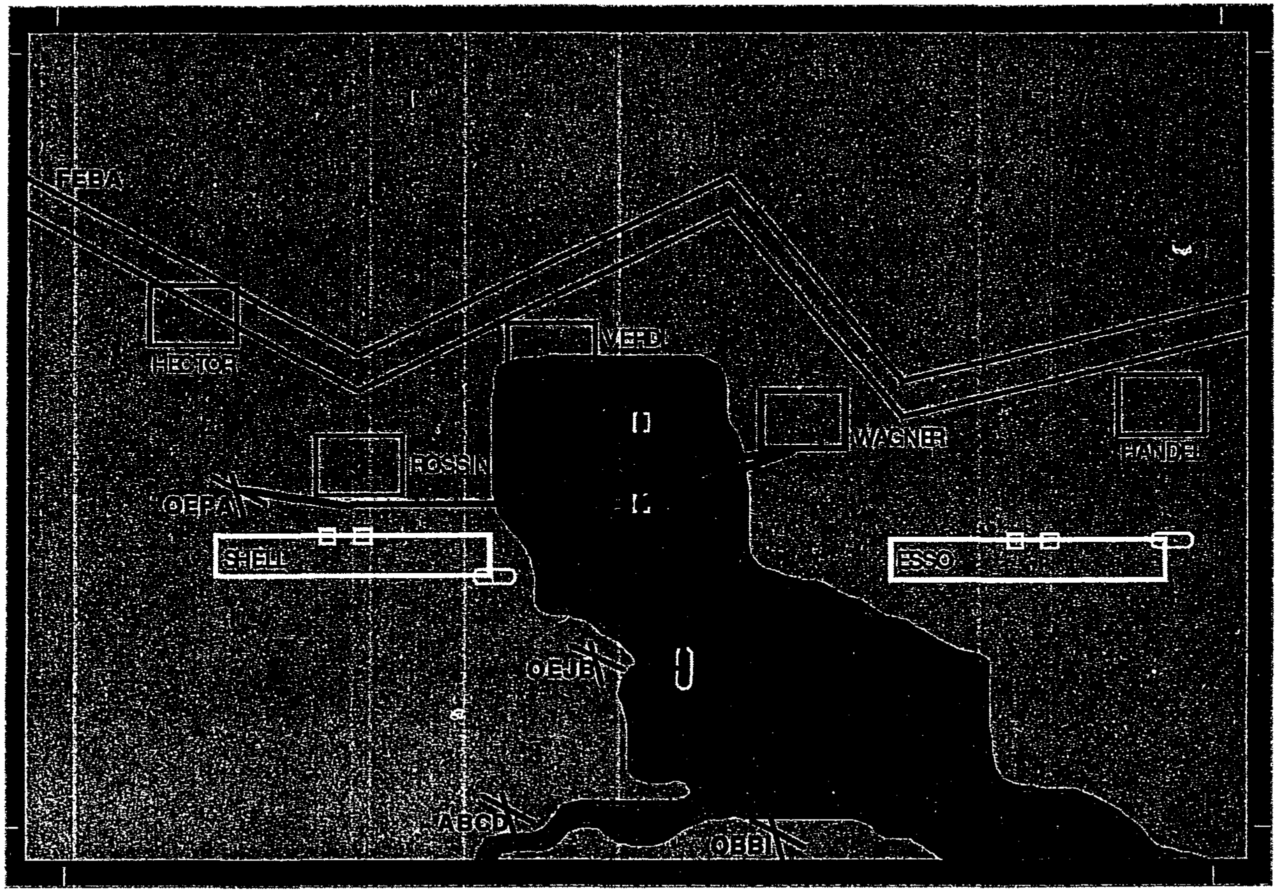
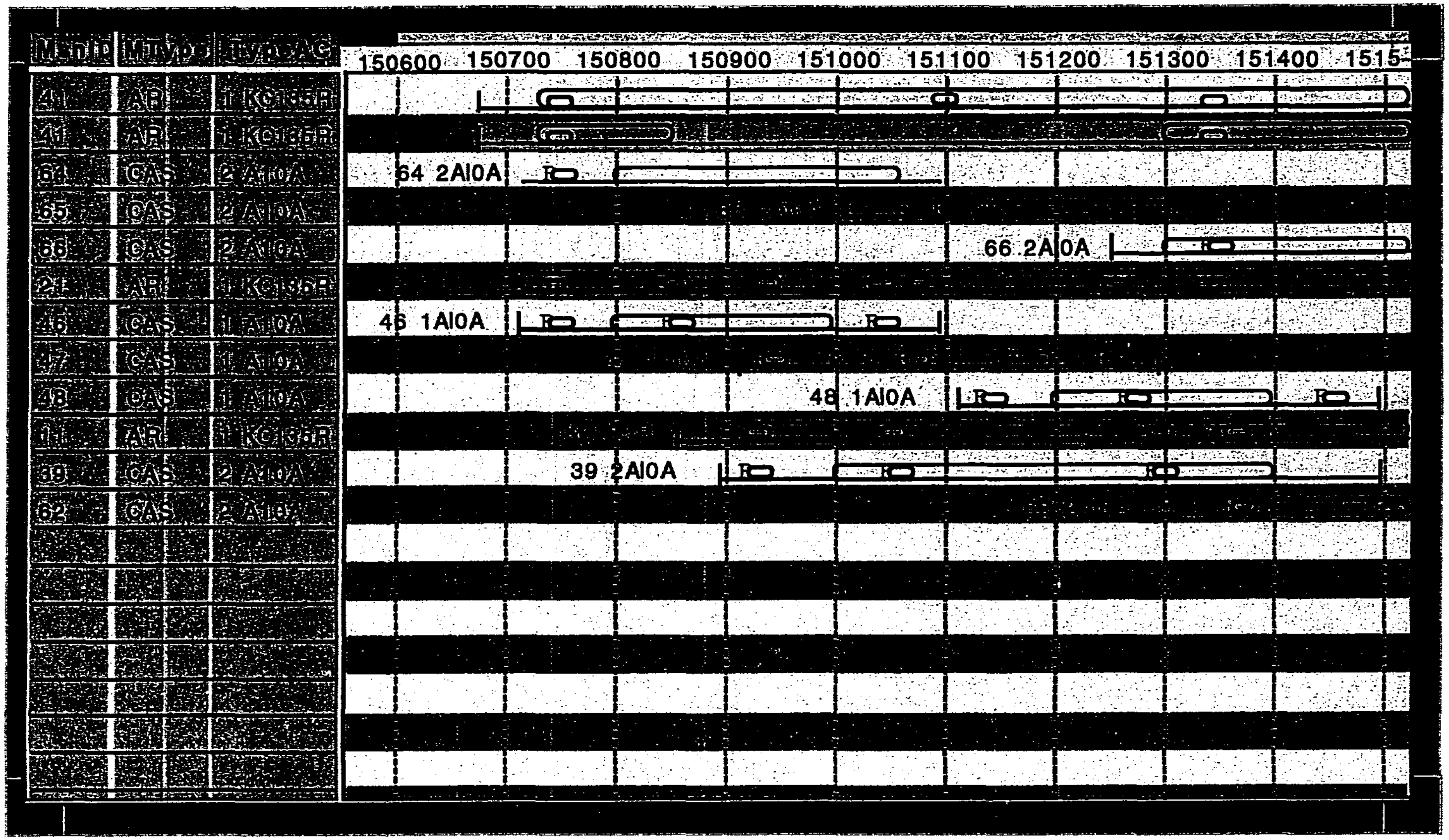
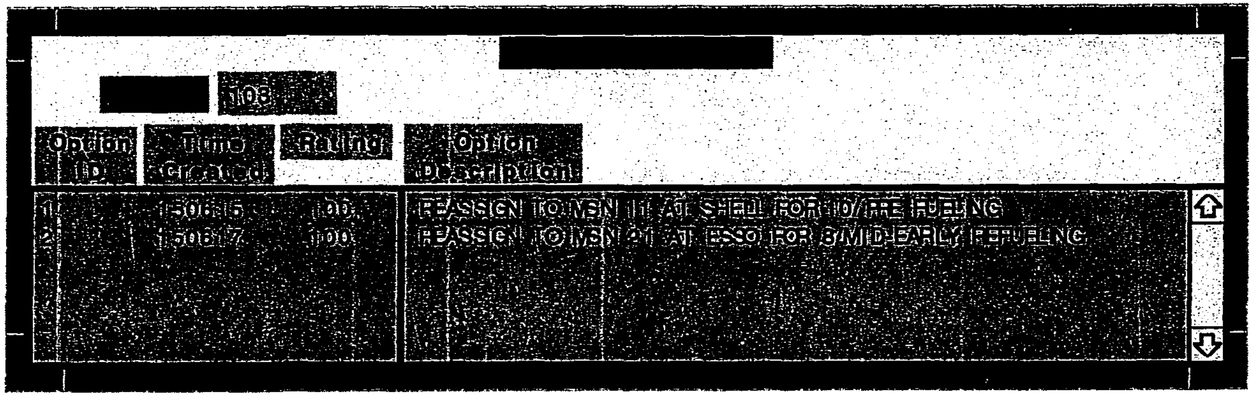


Figure G-5: Map Graphic
FLEX Window (Modified for Case Study)



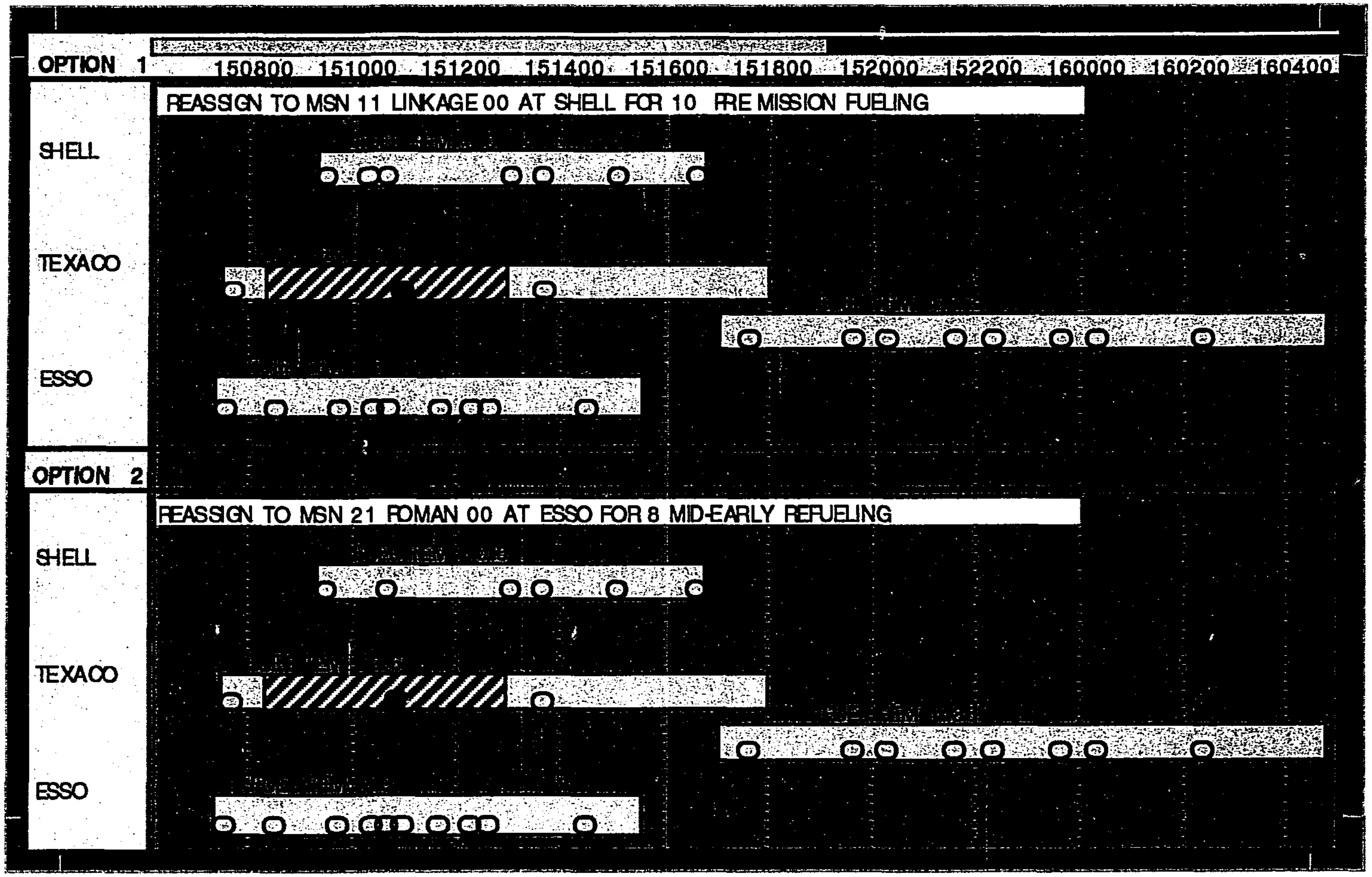
MsnID	MType	TypeAC
41	AR	1 KC135R
41	AR	1 KC135R
61	CAS	2 A10A
65	CAS	2 A10A
66	CAS	2 A10A
21	AR	1 KC135R
46	CAS	1 A10A
47	CAS	1 A10A
48	CAS	1 A10A
11	AR	1 KC135R
39	CAS	2 A10A
62	CAS	2 A10A

Figure G-6: Marquee
FLEX Window (Modified for Case Study)



Option ID	Time Created	Rating	Option Description
1	150615	100	REASSIGN TO MSN 11 AT SHELL FOR 10/PRE FUELING
2	150617	100	REASSIGN TO MSN 21 AT ESSO FOR 8/MID-EARLY REFUELING

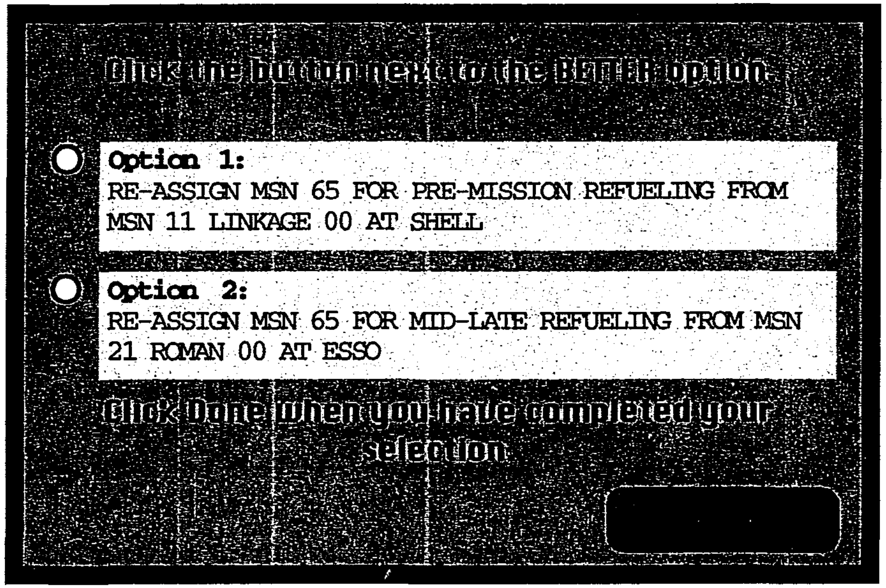
Figure G-7: Replanning Options
FLEX Window (Modified for Case Study)



REASSIGN TO MSN 11 LINKAGE 00 AT SHELL FOR 10 PRE MISSION FUELING

REASSIGN TO MSN 21 ROMAN 00 AT ESSO FOR 8 MID-EARLY REFUELING

Figure G-8: Option View
Cognitively Engineered Window - Support to Replanning Option Evaluation for Tanker Operations



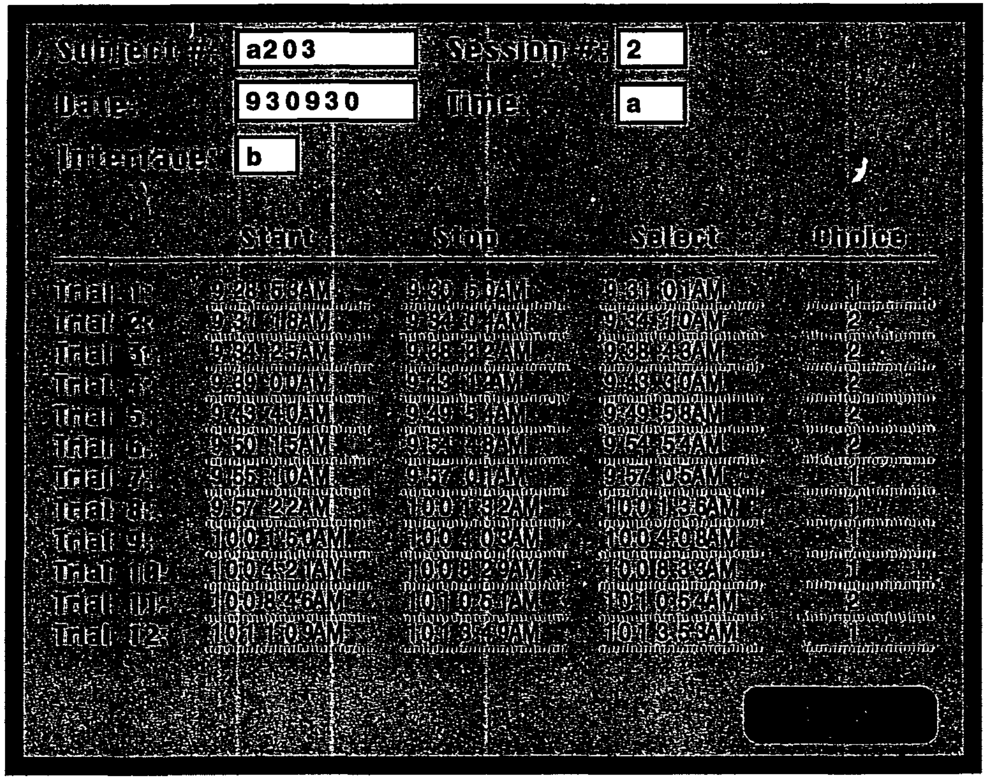
Click the button next to the BETTER option.

○ **Option 1:**
RE-ASSIGN MSN 65 FOR PRE-MISSION REFUELING FROM MSN 11 LINKAGE 00 AT SHELL

○ **Option 2:**
RE-ASSIGN MSN 65 FOR MID-LATE REFUELING FROM MSN 21 ROMAN 00 AT ESSO

Click Done when you have completed your selection.

Figure G-9: Option Select
Automated Data Collection #1: Capturing Decision



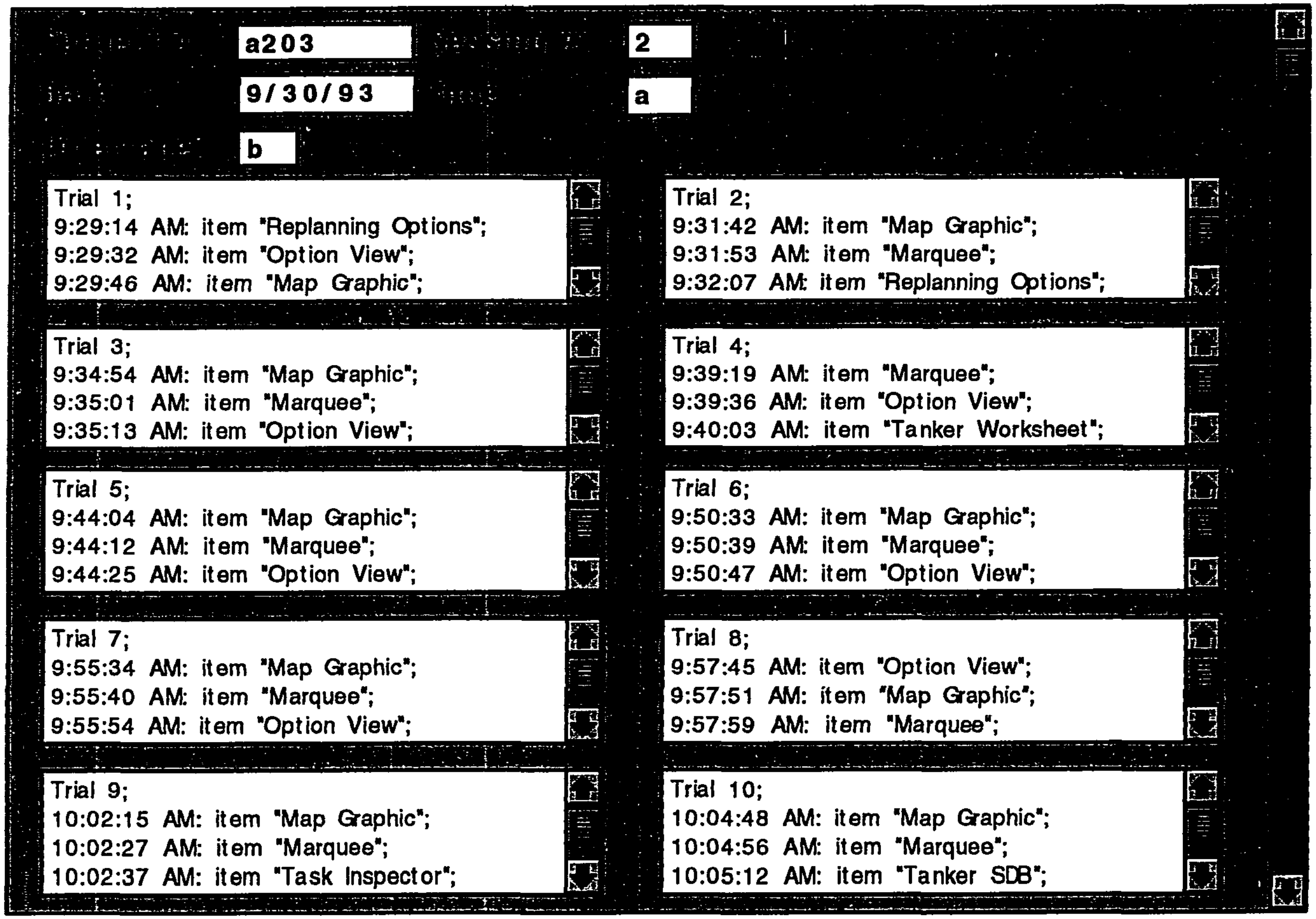
Subject #: a203 Session #: 2

Date: 930930 Time: a

Interface: b

	Start	Stop	Select	Choice
Trial 1:	9:28:58AM	9:30:50AM	9:31:01AM	1
Trial 2:	9:31:18AM	9:34:04AM	9:34:10AM	2
Trial 3:	9:34:25AM	9:38:32AM	9:38:43AM	2
Trial 4:	9:39:00AM	9:43:12AM	9:43:30AM	2
Trial 5:	9:43:40AM	9:49:54AM	9:49:58AM	2
Trial 6:	9:50:15AM	9:54:48AM	9:54:54AM	2
Trial 7:	9:55:10AM	9:57:01AM	9:57:05AM	1
Trial 8:	9:57:22AM	10:01:32AM	10:01:36AM	1
Trial 9:	10:01:50AM	10:04:03AM	10:04:08AM	1
Trial 10:	10:04:21AM	10:08:29AM	10:08:33AM	1
Trial 11:	10:08:46AM	10:10:51AM	10:10:54AM	2
Trial 12:	10:11:09AM	10:13:49AM	10:13:53AM	1

Figure G-10: Subject Card
Automated Data Collection #2: Speed and Accuracy



a203

2

9/30/93

a

b

Trial 1;
9:29:14 AM: item "Replanning Options";
9:29:32 AM: item "Option View";
9:29:46 AM: item "Map Graphic";

Trial 2;
9:31:42 AM: item "Map Graphic";
9:31:53 AM: item "Marquee";
9:32:07 AM: item "Replanning Options";

Trial 3;
9:34:54 AM: item "Map Graphic";
9:35:01 AM: item "Marquee";
9:35:13 AM: item "Option View";

Trial 4;
9:39:19 AM: item "Marquee";
9:39:36 AM: item "Option View";
9:40:03 AM: item "Tanker Worksheet";

Trial 5;
9:44:04 AM: item "Map Graphic";
9:44:12 AM: item "Marquee";
9:44:25 AM: item "Option View";

Trial 6;
9:50:33 AM: item "Map Graphic";
9:50:39 AM: item "Marquee";
9:50:47 AM: item "Option View";

Trial 7;
9:55:34 AM: item "Map Graphic";
9:55:40 AM: item "Marquee";
9:55:54 AM: item "Option View";

Trial 8;
9:57:45 AM: item "Option View";
9:57:51 AM: item "Map Graphic";
9:57:59 AM: item "Marquee";

Trial 9;
10:02:15 AM: item "Map Graphic";
10:02:27 AM: item "Marquee";
10:02:37 AM: item "Task Inspector";

Trial 10;
10:04:48 AM: item "Map Graphic";
10:04:56 AM: item "Marquee";
10:05:12 AM: item "Tanker SDB";

Figure G-11: Subject Tracker
Automated Data Collection #3: Process Tracing

Appendix H:

FLEX Project Acronyms

FLEX Project Acronyms

2SFW	Two Line Sequential Flow
-A-	
AAA	Anti-Aircraft Artillery
AAC	Alaskan Air Command
AADC	Area Air Defense Commander
AAFCE	Allied Air Forces - Central Europe
AAGS	Army Air Ground System
AAR	Air to Air Refueling
AAS	Advanced Planning System Application Software
AAS	Aircraft/Aircrew Status
ABCCC	Airborne Command & Control Center
ABP	Air Battle Plan
AC	Aircraft
AC2SMAN	Alaskan Command & Control System Military Automated Network
ACA	Airspace Control Authority
ACC	Air Component Commander
ACC	Air Combat Command (formerly TAC)
ACCIS	(old, but still useful, source of intelligence)
ACE/OPS	Airborne Command Element/Current Ops

ACO	Air Space Coordination Order
ACS	Air Control System
ADA	Air Defense Artillery
ADRG	ARC Digitized Raster Graphics
ADRI	ARC Digitized Raster Imagery
ADW	Air Defense Weapons Status
AFAC	Airborne Forward Air Controller/Coordinator
AFARN	Air Force Air Request Net
AFEWC	Air Force Electronic Warfare Center
AFF	Airfield/Flight Facility Status
AFFOR	Air Force Forces
AFMC	Air Force Materiel Command
AFSC	Air Force Systems Command
AFSC	Air Force Specialty Code
AFWCCS	Air Force Wing Command and Control System
AI	Air Interdiction
ALBM	Air Land Battle Management
ALCC	Airlift Control Center
ALCE	Airlift Control Element
ALCOM	Alaskan Command
ALO	Air Liaison Officer
AME	Airspace Management Element
ANR	Alaskan NORAD Region
AOB	Air Order of Battle
AOC	Air Operations Center (formerly TACC)
APS	Advanced Planning System

AR	Aerial Refueling
ARC	(Equal) Arc Second Projection
ARCP	Aerial Refueling Contact Point
ARCT	Aerial Refueling Contact Time
ASDS	Air Situation Display System
ARPA	Advanced Research Projects Agency (formerly DARPA)
ASOC	Air Support Operations Center
ASOC SFW	ASOC Sequential Flow
ASO/T	Air Surveillance Officer/Technician
ATACC	Advanced Tactical Air Control Center
ATAF	Allied Tactical Air Force
ATC	Air Traffic Control
ATM	Air Tasking Message
ATO	Air Tasking Order
ATOC	Allied Tactical Operations Center
ATTD	Advanced Technology Transition Demonstration
AWACS	Airborne Warning & Control System
AWDS	Automated Weather Distribution System
AWN	Air Weather Network
AWS	Air Weather Service

-B-

BAI	Battlefield Air Interdiction
BASS	BCE Auto Support System
BCE	Battlefield Coordination Element
BDA	Battle Damage Assessment
BIM	Battle Information Management

BM	Battle Management
BMCL	Battle Management Concepts Laboratory

-C-

C2	Command & Control
C2TC	Command & Control Technology Center
C3	Command, Control & Communications
C3AA	Rome Laboratory, Command, Control & Communications Division, Advanced Concepts Branch (FLEX ATTD Program management)
C3AB	Rome Laboratory, Command, Control & Communications Division, Computer Systems Branch
C3CM	Command, Control & Communications Countermeasures
C3CM BMDA	Command, Control & Communications Countermeasures Battle Management Decision Aid
C3I	Command, Control, Communications & Intelligence
CA	Counter Air
CAFMS	Computer Assisted Force Management System
CAME	Corps Airspace Management Element
CAP	Combat Air Patrol
CAS	Close Air Support
CATSS	Cartographic Applications for Tactical & Strategic Systems
CBR	Chemical, Biological, Radiological
CCF	Commander's Constraint File
CCO	Chief of Combat Operations
CCS	Communications Circuit Status
CCT	Combat Control Team

CCTL	Command & Control Test Laboratory
CE	Control Element
CENTAF	Central Tactical Air Force
CENTCOM	Central Command
CIAD	Combat Operations Intelligence Division (PACAF equivalent to CID)
CID	Combat Intelligence Division
CIJ	Close In Jamming
CMP	Common Mapping Program
CMS	Common MCG&I System
COA	Combat Operations Automation
COD	Combat Operations Division
CODS	Combat Operations Decision Support
COID	Combat Intelligence Application Division (PACAF equivalent to ENSCD)
COMAO	Combined Air Operations
COMINT	Communications Intelligence
COMSEC	Communications Security
COSMOS	Cost/Schedule Management-Oriented System
COTS	Commercial Off-the-Shelf
CPD	Combat Plans Division
CRC	Control & Reporting Center
CRP	Control & Reporting Post
CTAPS	Contingency Tactical Air Control System Automated Planning System
CTOC	Corps Tactical Operations Center
CW	Constant Watch Program

CW/OIA	Constant Watch/Operations Intelligence Automation

-D-

DAFIF	Digital Aeronautical Flight Information File
DARPA	Defense Advanced Research Projects Agency (now ARPA)
DASC	Direct Air Support Center
DBMS	Data Base Management System
DCA	Defensive Counter-Air
DCA	Digital Cartographic Applications
DCS	Defensive Communications System
DCT	Digital Communications Terminal
DCW	Digital Chart of the World
DEC	Digital Equipment Corporation
DFAD	Digital Feature Analysis Data
DLMS	Digital Landmass System
DMA	Defense Mapping Agency
DMPI	Desired Mean Point of Impact
DMS	Data Management System
DMTD	Digital Map Terrain Data
DO	Duty Officer (Combat Operations - ex.: F-15 DO)
DoD	Department of Defense
DOS	Disk Operating System
DP	Design Prototype
DTED	Digitized Terrain Elevation Data
DVOF	Digital Vertical Obstruction Data

-E-

EAC	Emergency Action Cell
EC	Electronic Combat
ECHUM	Electronic Chart Update Manual
ECCM	Electronic Counter-Countermeasures
ECM	Electronic Countermeasures
ED	Enemy Destruction
EIFEL	Elektronisches Informations und Fuhrung System fur die Einsatzbereitschaft fur der Luftwaffe
ELINT	Electronic Intelligence
ENSCD	Enemy Situation Correlation Division
ENSCE	Enemy Situation Correlation Element
EO	Evaluation Officer/Early Warning/Intelligence
EOB	Enemy Order of Battle
EOB	Electronic Order of Battle
EP	Evaluation Prototype
ERA	Entity-Relationship-Attribute
ESC	Electronic Systems Center
ESD	Electronic Systems Division (now ESC)
ESM	Electronic Support Measures
ETOT	Estimated Time Over Target
EW	Electronic Warfare

-F-

FAC	Forward Air Controller
FACP	Forward Air Control Post
FACS	Feature Attribute Coding System

FAOR	Fighter Area of Responsibility
FEBA	Forward Edge of Battle Area
FEM	Field Evaluation Model
FF	Fighter Flow (ASOC station)
FFIRN	Field Format Index Reference Number
FIPS	Federal Information Processing Standard
FLAPS	Force Level Automated Planning System
FLEX	Force Level Execution
FLET	Forward Line of Enemy Troops
FLOT	Forward Line of Own Troops
FM	Frequency Modulation
FMS	Fighter Mission Schedule
FOB	Friendly Order of Battle
FSCL	Fire Support Coordination Line
FSE	Fire Support Element
FTP	File Transfer Protocol
FUD	Field Use Designator

-G-

G-2	Army Intelligence Staff
G-3	Army Operations Staff
GACC	Ground Attack Control Center/Capability
GAT	Guidance, Apportionment, Targeting
Gb	Gigabyte
GE	Ground Environment
GENSER	General Service
GEOREF	(World) Geographic Reference System

GFAC	Ground Forward Air Controller
GLO	Ground Liaison Officer
GLT	Ground Liaison Team (subdivision of ASOC)
GOB	Ground Order of Battle
GOSIP	Government Open Systems Interconnect Profile
GUI	Graphical User Interface
GWC	Global Weather Center

-H-

HF	High Frequency
HOFEZ	Hostile Fire Engagement Zones
Hq	Headquarters
HTACC	Hardened Tactical Air Control Center
HUMINT	Human Intelligence

-I-

ICAO	International Civil Aviation Organization
ICORD	Initial Combat Operations Replanning Design
ID	Identifier/Identification
IFF	Identification Friend or Foe
IKP	Instructor/Key Personnel
ILS	Integrated Logistics Support
ILSP	Integrated Logistics Support Plan
IMOM	Improved Many-on-Many Model
INT	Air Interdiction
I/O	Input/Output
IP	Internet Protocol

IR	Infrared
IR&D	Independent Research & Development
ISO	International Standards Organization
ISP	Integrated Support Plan
ITACC	Integrated Tactical Air Control Center
ITD	Interim Terrain Data

-J-

JAAT	Joint Air Attack Team
JAMPS	JINTACCS Automated Message Preparation System
JCS	Joint Chiefs of Staff
JINTACCS	Joint Interoperability of Tactical Command & Control Systems
JMEM	Joint Munitions Effectiveness Manual
JSEAD	Joint Suppression of Enemy Aid Defenses
JTF	Joint Task Force
JTS-AK	Joint Task Force Alaska

-K-

K	Thousand
Kb	Kilobytes

-L-

LAN	Local Area Network
LAT	Latitude
LCC	Life Cycle Cost
LENSCE	Limited Enemy Situation Correlation Element
LLTR	Low Level Transit Route

LOCE	Limited Operational Capability Europe (US developed gateway for NATO access to fused intelligence sources)
LONG	Longitude
LOS	Line of Sight
LSA	Logistics Support Analysis
LSAP	Logistics Support Analysis Plan

-M-

MAJCOM	Major Air Force Command
Mb	Megabyte
MCG&I	Mapping, Charting, Geopositioning, & Imagery
MEZ	Medium Engagement Zones
MGRS	Military Grid Reference System
MISREP	Mission Report
MMI	Man-Machine Interface
MMW	Millimeter Wave
MOB	Missile Order of Battle
MOTIF	Open System Foundation User Interface Standard
MPC	Message Processing Center
MRR	Minimum Risk Route
MSL	Mean Sea Level
MSNCC	Mission Commander
MSS	Mission Support System
MSS	Munitions Status
MTBF	Mean Time Between Failure
M/T	Maintainability/Testability

-N-

NAD	North American Datum
NALE	Naval & Amphibious Liaison Element
NATO	North Atlantic Treaty Organization
NBC	Nuclear, Biological, Chemical
NOB	Naval Order of Battle
NUCINT	Nuclear Intelligence

-O-

OAS	Offensive Air Support
OB	Order of Battle
OCA	Offensive Counter-Air
OSI	Open Systems Interconnect
OTDA	Operational Tactical Decision Aid

-P-

Paa	Probability of Arrival
PACAF	Pacific Air Force
PCA	Physical Configuration Audit
Pd	Probability of Destruction
PGM	Precision Guided Munitions
PHS&T	Packaging, Handling, Storage & Transportation
Pk	Probability of Kill
POL	Petroleum, Oil & Lubricant
PMO	Program Management Office
PRS	Procedural Reasoning System
Ps	Probability of Success/Survival

PVOD	Probabilistic Vertical Obstruction Data
-Q-	
QA	Quality Assurance
-R-	
R & S	Reconnaissance & Surveillance
RAAP	Rapid Application of Air Power
RADC	Rome Air Development Center (now RL for Rome Laboratory)
RADINT	Radar Intelligence
REC	Reconnaissance
RECCE	Reconnaissance
RL	Rome Laboratory
RMS	Reconnaissance Mission Schedule
ROE	Rules of Engagement
ROZ	Restricted Operating Zone
-S-	
SAC	Strategic Air Command
SADO	Senior Air Defense Duty Officer
SAM	Surface to Air Missile
SAMCO	SAM Coordinator
SAR	Search & Rescue
SATCOM	Satellite Communications
SCD	Selection Criteria Display
SCI	Sensitive Compartmented Information

SCL	Standard Conventional Load
SDB	Status Display Board
SEAD	Suppression of Enemy Air Defenses
SEC CON	Sector Control (coordinates SAMCO, FF, ASO/T, & EO stations in ASOC)
SIF	Selective Identification Feature
SMS	Softcopy Mapping System
SOC	Sector Operations Center
SODO	Senior Operations Duty Officer
SOJ	Stand-Off Jamming
SPINS	Special Instructions
SQL	Structured Query Language
SR	Status Report
SRS	System Requirement Specification
SSS	System/Segment Specification

-T-

TAC	Tactical Air Command (Air Combat Command (ACC) after 1 June 92)
TACC	Tactical Air Control Center (Air Operations Center (AOC) after 1 June 92)
TACP	Tactical Air Control Party
TACR	Tactical Air Command Regulation
TACS	Theater Air Control System
TACS	Tactical Air Control System
TACS DEMO	Distributed TACS Data Base Demonstration Program
TADIL	Tactical Air Data Information Link

TAF	Tactical Air Force
TAFIIS	Tactical Air Forces Integrated Information System
TAI	Target Area of Interest
TAL	Tactical Airlift
TALO	Tactical Airlift Liaison Officer
TAR	Tactical Air Reconnaissance
TAT	Turn Around Time
TBM	Theater Battle Management
TDA	Tactical Decision Aid
TEMPEST	Emission Control Program
TEMPLAR	Tactical Expert Mission Planner
TERPES	Tactical Electronic Reconnaissance Processing & Evaluation
TFS	TACS Facility Status
Tgt	Target
TIS	Tactical Intelligence Squadron
TMS	Tactical Mapping System
TNL	Target Nomination List
TOC	Tactical Operations Center
TOT	Time Over Target (Time On Target)
TPFDL	Time Phased Force Deployment Listing
TPT	Training Plan Team
TPW	Target Planning Worksheet
TPWG	Test Plan Working Group
TRACE	Tactical Resource Allocation Control Element
TRI-TAC	Tri-Service Tactical Communications
TTY	Teletype

-U-

UHF	Ultra High Frequency
UCI	User-Computer Interface
UI	User Interface
UIM	User Interface Management
UK	United Kingdom
US	United States
USAF	United States Air Force
USAFE	United States Air Force Europe
USGS	United States Geodetic Services
USMTF	United States Message Text Format
UTM	Universal Transverse Mercator

-V-

VHF	Very High Frequency
VMS	Variable Mission Schedule

-W-

WCCS	Wing Command & Control System
WGS	World Geodetic System
WML	Working Mission Line
WOC	Wing Operations Center
WSS	Weather Status
WVS	World Vector Shoreline
WWMCCS	World Wide Military Command & Control System

Appendix I:

Summary of Change Requests

Summary of Change Requests
(as of 20 Oct 93)

PREF ID VERSION ID DISPOSITION SUBJECT

DESCRIPTION ____

RESPONSE ____

RL 1 FAAS DEMO1 PROTO 3 ATTACK MISSION STATUS REPORT

Remove the Retrieve Data from Database button and do the query automatically when you enter the MissionID and/or Callsign.

The query button placed next to mission id and callsign field. Status report is automatically populated when popped up from a status display board or marquee.

RL 2 FAAS DEMO1 PROTO 3 ATTACK QUERY

It might be better to remove the word "ATTACK" and have more available defaults for all missions, ie recce, defense, AAR, etc.

There is now only one SDB, with mission subset queries available (i.e. tanker, attack).

RL 3 FAAS DEMO1 PROTO 2 ACTION REQUEST LIST

Combine with Status_Report_List for a single more informative window.

O.B.D. Action Requests changed to tasks and redesigned.

RL 4 FAAS DEMO1 6.3B LOGIN SCREEN

Ability to name positions by misson type or A/C, use NATO mission types.

Login positions will need to be addressed during CTAPS integration since the Login will be through CTAPS, not FLEX. This is also a theater tailoring issue.

RL 5 FAAS DEMO1 PROTO 3 MARQUEE

Identify day/night on time bar wiith different colors, (e.g. yellow for day, grey for nigh).

Added bar above time bar with yellow for day and black for night.

RL 6 FAAS DEMO1 PROTO 3 MARQUEE

It would be useful to make changes directly on marquee. e.g. click and drag to new TOT, ARCT, etc. Then show what conflicts may be generated.

Hot keys have been implemented which allow the user to cancel, divert, and launch a mission directly from the marquee. However, clicking and dragging icons to produce changes will only work if the system is redesigned to allow better communication between the UI, ABM, and DB.

RL 7 FAAS DEMO1 N/A USER INTERFACE

Change data entry functionality from "Insert" to "Overwrite" (i.e., when you highlight a field, cursor is at beginning of the field and typing will overwrite entry. If you just hit return, it will accept current contents and move to beginning of the next field.)

Overwriting capability within Motif was investigated and not a feature of text widgets at this time. A return will accept current contents and go to next field. As newer versions of MOTIF come on line this issue should be addressed again.

RL 8 FAAS DEMO1 PROTO 2 STATUS REPORT LIST

Combine with Action Request List for single, more informative window.

O.B.D. Action Requests have been changed to tasks and redesigned.

RL 9 FAAS DEMO1 PROTO 3 STATUS DISPLAY BOARDS

Have single status display board with all fields available for all missions. Then have more default queries available.

Implemented.

RL 10 FAAS DEMO1 6.3 B LOGIN SCREEN

User definable or theater tailored names.

This request will be addressed in the FLEX 6.3b effort. It will be addressed as a part of integration to CTAPS since the login will be through CTAPS not FLEX. This is also a theater tailoring issue.

RL 11 FAAS DEMO1 6.3 B STATUS DISPLAY BOARDS

Highlight air borne sorties.

Needs management approval to do.

RL 12 FAAS DEMO1 PROTO 2 ATTACK QUERY

Add "Target Info" as a queryable field: query on target, target group, or target type. (may require another window)

You can now query on target, target type, but not target group.

RL 13 FAAS DEMO1 PROTO 2 ACTION REQUEST LIST

List DOs who will need to coordinate with AR. Show which ones have not looked at it yet so that I can track who needs my attention/help.

Tasking implementation shows all subtasks, who they are assigned to, and their status.

RL 14 FAAS DEMO1 PROTO 2 USER INTERFACE

Flag to alert the SODO as soon as all DOs have coordinated on an AR (or have completed all sub ARs). I need to know when to review and approve the action.

Subtask returned window is shown when the subtask is ready for approval.

RL 15 FAAS DEMO1 PROTO 3 ACTION REQUEST

Show limited mission data somewhere: ETD, TOT, Unit, Type A/C, etc.

Shown in conflicts and plan changes list.

RL 16 FAAS DEMO1 PROTO 2 MARQUEE

Highlight air borne missions.

Mission indicators are drawn in green to show estimates and in blue to show actual.

RL 17 FAAS DEMO1 N/A MARQUEE

Capability to vertically compress (ie. show 2 times or even 4 times as many missions but with less detail). SODO needs a view of the entire day, even if it has to be hard copy on a big sheet.

This would be a nice feature but is expensive to implement since text would have to be scalable.

RL 18 FAAS DEMO1 PROTO 2 TARGET STATUS REPORT

Query to show unassigned targets.

The query that will show unassigned targets is selecting a target with zero missions scheduled.

RL 19 FAAS DEMO1 6.3 B MARQUEE

Add a print capability.

Besides screen captures, MOTIF does not support generating postscript files of windows to print graphics.

RL 20 FAAS DEMO1 PROTO 2 USER INTERFACE

Combine AR List, AR, Status Report List, and Status Report.

O.B.D. Action requests have been changed to tasks and redesigned.

RL 21 FAAS DEMO1 6.3 B LOGIN SCREEN

Add liason, USN and USMC. Add SAR, SOF, AIRLIFT.

Login positions will need to be addressed during CTAPS integration since the login will be through CTAPS not FLEX.

RL 22 FAAS DEMO1 6.3 B STATUS DISPLAY BOARDS

Would like to see some standard sorts like CAFMS: 2 line ASOC, 2 line Tanker, etc.

RL 23 FAAS DEMO1 6.3 B STATUS REPORTS

By clicking on a field the system should give a read out of what the fields values are, i.e. PKG would list all a/c in the package or associated with this mission. This function is found in CAFMS, query a field for a set of possible values.

The query screens for proto 3 demonstrate this capability for 4 fields that can be queried for a set of possible values with the right mouse button. This can be easily extended to status reports.

RL 24 FAAS DEMO1 6.3 B MARQUEE

Sort by area: North-Central, etc.

Not possible to sort on direction, may be possible after a redesign of the DB.

RL 25 FAAS DEMO1 PROTO 3 STATUS REPORTS

Include A/R times with TOT/TFT, Take off, and landing times.

Air refueling is now displayed on attack status report.

RL 26 FAAS DEMO1 6.3 B TANKER STATUS DISPLAY BOARD

Show receivers upon demand, include mission numbers and TOTs.

Will be done as part of SDB bundling.

RL 27 FAAS DEMO1 PROTO 3 USER INTERFACE

Put ATO ID on the windows. i.e. status reports and SDBs

ATO ID is displayed on the flex main menu which should always be visible.

RL 28 FAAS DEMO1 PROTO 2 MARQUEE

Add vertical/horizontal line cursor that the operator can toggle off and on.

O.B.D. Redesign of the marquee look and feel made
this obsolete.

RL 29 FAAS DEMO1 PROTO 2 MARQUEE

Highlight a line consisting of the three columns of information on the left along with the graphical icon.

Implemented.

RL 30 FAAS DEMO1 PROTO 2 ACTION REQUEST

1 AR where everyone initials off on it (concerned about Sub-ARs proliferating). SODO wants to know current status. Coordination for approval

O.B.D. Tasking implementation does this.

RL 31 FAAS DEMO1 6.3 B STATUS DISPLAY BOARDS

Put a 132 character column marker on the SDBs. To tell how much can be printed.

RL 32 FAAS DEMO1 PROTO 3 MARQUEE

Show relationships/dependancies between missions to 1 level.

Bundling...

Bundling and Emphasize implementations.

RL 33 FAAS DEMO1 PROTO 3 MARQUEE

Add callsign, iff, msn type, and target to information lists.

Made information lists the same as the sort lists.

RL 34 FAAS DEMO1 6.3 B UIM/MAP

Query on the Status Display Board shows up on the map also.

RL 35 FAAS DEMO1 PROTO 3 QUERY

Query screens should come up automatically with the previous query.

Previous query now remains.

RL 36 FAAS DEMO1 PROTO 2 ACTION REQUEST

Action Request should be renamed.

Renamed to task.

RL 37 FAAS DEMO1 6.3 B QUERY

Would like to query based on geographical area, according to lat/long or UTM.

RL 38 FAAS DEMO1 PROTO 2 STATUS DISPLAY BOARDS

Instead of having the P/E/A columns, have them all seperate and let the user tailor them out.

They are now seperate columns and they are tailorable.

RL 39 FAAS DEMO1 6.3 B ALERTS

When the user is alerted of some problem, something needs to be displayed on the Marquee/SDBs to indicate this.

RL 40 FAAS DEMO1 PROTO 3 MARQUEE

Unit flow functionality of APS shows how a problem with one mission can affect that unit. May need that functionality as a popup from the marquee, or integrated into it.

APS unit flow functionality added to marquee.

RL 41 FAAS DEMO1 6.3 B REPEAT CALLSIGNS, MISSION IDS

There may be repeat callsigns (and with 'seamless' ATO, there may be repeat mission IDs), so will need to default to the closest (current), with a next button.

RL 42 FAAS DEMO1 6.3 B URGENCY

Urgency may need to be user-tailorable, with weighted inputs, etc, used to calculate the urgency. Also, urgency needs to change as time goes on. As TOT approaches, urgency increases.

RL 43 FAAS DEMO1 PROTO 2 STATUS DISPLAY BOARDS

Need to query on mission type and time.

You can now query on Mission type. You can not query on times but this is addressed under User comment number 83.

RL 44 FAAS DEMO1 6.3 B STATUS REPORT

For SOURCE field, would like an option menu , plus the ability to type something else in.

This can be easily implemented using the same design as the right bottom popup on query screens. A list of frequent sources should be entered into the data base in theater.

RL 45 FAAS DEMO1 6.3 B RESULTS ENUMERATION

Need to review the CAFMS result field enumeration, as possible values for FLEX.

An extension of this problem will need to be addressed in 6.3b, i.e., how FLEX can interpret the results. e.g. should a particular result trigger a new task.

RL 46 FAAS DEMO1 PROTO 3 ACTION REQUEST

AR and AR flag should tell the user more about the problem, what happened.

Tasking implementation with plan changes and conflicts also subtask returned and task notify windows contain extra information.

RL 47 FAAS DEMO1 PROTO 2 MARQUEE

Would like to be able to select one of the columns displayed on the marquee, and be able to Sort Ascending/ Descending, similar to APS.

Sort has been put on the marquee with ascending and descending options.

RL 48 FAAS DEMO1 6.3 B TARGET STATUS REPORT

Should there be a way of indicating the input of a new target on the Target SR.

RL 49 FAAS DEMO1 PROTO 2 ACTION REQUEST QUERY

Include name and maybe mission type as queryable fields.

Many queriable fields were added.

RL 50 FAAS DEMO1 PROTO 2 CUT AND PASTE

Users should be able to cut and paste within FLEX and between other applications.

Cut and paste capability for text is available between all MOTIF applications.

RL 51 FAAS DEMO1 PROTO 3 USER INTERFACE

Be careful about the number of screens developed. It can get confusing with a lot of small screens.

Redesigned to combine and eliminate multiple screens.

RL 52 FAAS DEMO1 PROTO 3 ACTION REQUEST

Action Request Status field should include Acknowledged.

One of the status values is returned, indication the user has acknowledged the task.

RL 53 FAAS DEMO1 6.3 B PLANNING WORKSHEET

Mission should be highlighted when moving from an Action Request to a planning worksheet.

Need to redesign planning worksheets.

RL 54 FAAS DEMO1 PROTO 3 AUTOMATIC TIME GENERATION

If the estimated TOT is changed in a status report, the estimated take off and landing time should be automatically updated as it is in APS.

Implemented.

RL 55 FAAS DEMO1 PROTO 2 MAIN MENU

If a window is iconified, it should reappear at the front of the screen when selected from a menu or window (APS does this but FLEX does not).

Implemented for Prototype 2.

RL 56 FAAS DEMO1 PROTO 2 MARQUEE

Include "mission type" as a column selection on marquee.

Included.

RL 57 FAAS DEMO1 N/A MAIN MENU

If the top level menu has only one selection, the mouse has to be clicked twice. One click should be enough.

Will not do. It goes against MOTIF conventions.

RL 58 FAAS DEMO1 PROTO 3 QUERY WINDOWS

Query window needs to disappear when the apply button is selected.

Done. Disappear s upon apply.

RL 59 FAAS DEMO1 PROTO 2 ACTION REQUEST

Too many windows ... gets confusing.

Tasking implementation uses fewer windows.

RL 60 FAAS DEMO1 PROTO 2 SDB

Put TRP column next to package column on SDB.

Done. TRP next to PKG NM.

RL 61 FAAS DEMO1 PROTO 2 STATUS REPORTS

Eliminate need for a <cr> when entering a textfield value.

Implemented for prototype 2.

RL 62 FAAS DEMO1 6.3 B STATUS DISPLAY BOARDS

Marker for information changed from the plan.

RL 63 FAAS DEMO1 PROTO 3 STATUS DISPLAY BOARD

Store column and query defaults.

Done. Defaults for query and tailor.

RL 64 FAAS DEMO1 PROTO 3 TARGET STATUS REPORT

Show flow of missions against a target for the day.

Can query on a target on status display board to see a list of missions for that target, (must select multiple targets toggle)

RL 65 FAAS DEMO1 6.3 B USER INTERFACE

Let the user define a time window so that the lists will not get too long, i.e. 2 hour old ARs go away.

Currently can not query on creation time.

RL 66 FAAS DEMO1 6.3 B LOGIN SCREEN

Would like to have a font selection similar to CTAPS.

Login issues will need to be addressed during CTAPS integration since the login will be through CTAPS not FLEX.

RL 67 FAAS DEMO1 6.3 B QUERY

Would like to see both lat/long and UTM as fields on the SDB and let the user pick which one he'd like to see.

RL 68 FAAS DEMO1 PROTO 2 ACTION REQUEST

Would like to be able to select the action request window from the action request flag.

Implemented.

RL 69 FAAS DEMO2 PROTO 3 TARGET PLANNING WORKSHEET

Only add a mission number if resources are added. If re-tasking a current mission, keep mission number.

Implemented.

RL 70 FAAS DEMO2 6.3 B REPLANNING

Frequently performed tasks such as assigning a new target to a mission should have a minimum number of steps.

While Alerts, Cancel, and Divert have been added to the marquee, full satisfaction of this comment will not come until 6.3 B.

RL 71 FAAS DEMO2 6.3 B PROCESS

A help capability could be tutorial-like, that is lead the
user from one step in the process to the next. For example, ask "What do you want to do? A or B or C.

RL 72 FAAS DEMO2 PROTO 2 STATUS DISPLAY BOARDS

Two different types of sorts were demonstrated on the SDBs - one APS like and the other like the sort on the marquee. Users wanted both capabilities and did not seem to think it would be confusing.

Implemented.

RL 73 FAAS DEMO2 PROTO 3 STATUS DISPLAY BOARDS

For SDBs the fields callsign, type and number A/C, base and unit should be the leftmost columns, since they will always be referenced with mission number.

Done. Columns were reordered.

RL 74 FAAS DEMO2 PROTO 3 STATUS DISPLAY BOARDS

When a query is first done on a SDB it should be displayed on the UI sorted by mission ID.

Implemented.

RL 75 FAAS DEMO2 PROTO 3 STATUS DISPLAY BOARDS

On SDBs, an indication of the total number of missions should be displayed at base of mission column number.

Will have total number of missions.

RL 76 FAAS DEMO2 PROTO 3 MARQUEE

When marquee is initially queried it should be displayed on the UI sorted by TOT.

Implemented.

RL 77 FAAS DEMO2 PROTO 3 MARQUEE

Button on marquee should be called "Mission Information" instead of "Textual Information".

Implemented.

RL 78 FAAS DEMO2 PROTO 3 MARQUEE

A lookup capability should exist for "scl" and "base".
Possibly a third mouse button in the text field would pop up this information.

Done on query windows for some fields.

RL 79 FAAS DEMO2 PROTO 3 MARQUEE

On marquee's textual information window, when a tanker is referenced ("SHELL") display its callsign also.

Implemented.

RL 80 AAS DEMO2 6.3 B MARQUEE

User may want the capability of viewing more than one "Textual Information" window at a time.

Need to make windows dynamically created.

RL 81 FAAS DEMO2 PROTO 2 SEARCH

Search capability should not be case sensitive.

Implemented.

RL 82 FAAS DEMO2 PROTO 3 SORT

A record of the last sorting fields should be available, either on the monitor window or left on the sorting window the next time it is brought up.

Implemented.

RL 83 FAAS DEMO2 6.3 B QUERY, SORT, SEARCH

Be able to query, sort, search on time ranges, such as tanker station times.

RL 84 FAAS DEMO2 6.3 B QUERY

Be able to query two items in one field, e.g. query both OCA and INT.

RL 85 FAAS DEMO2 PROTO 3 SORT

Have the sort window go away when the sort is applied.

The sort window no longer exists.

RL 86 FAAS DEMO2 6.3 B QUERY

Need meta-Q to re-query.

RL 87 FAAS DEMO2 PROTO 2 MARQUEE

On marquee, always show planned times, and then either estimated or actual.

Implemented.

RL 88 FAAS DEMO2 PROTO 3 STATUS REPORT

Possibly have a "bad" mission result (aborted, etc) automatically default to mission status of UNSUCCESSFUL.

Aborted and cancelled missions automatically default
to "unsuccessful".

RL 89 FAAS DEMO2 PROTO 3 STATUS DISPLAY BOARDS

On SDBs, have button for "Show Unique", which would eliminate multiple rows per mission.

This has been implemented for Attack, EC, and Tanker missions.

RL 90 FAAS DEMO2 6.3 B LOGIN SCREEN

On login, would need multiple DOs (e.g. OCA1, OCA2, etc.).

This needs to be addressed during CTAPS integration.

RL 91 FAAS DEMO2 N/A STATUS REPORT

Would like to specify which D.O. a task will go to when an SR is input.

Not possible when submitted electronically.

RL 92 FAAS DEMO2 6.3 B PROCESS

Will eventually need to send targets untasked back to RAAP/ APS.

Problem will be addressed at Proto 3.

RL 93 FAAS DEMO2 6.3 B CHANGE TASK ORDER

Disseminate the CTO: the user will need to be able to designate locations to be sent to (beside just the defaults) Will also need to be able to send a CTO to "no one" if the necessary coordination is done over the phone.

RL 94 FAAS DEMO2 PROTO 3 USER INTERFACE

Need more indications of what the system is doing e.g. working windows.

We now have information on the marquee, and we make use of the stopwatch cursor to indicate when queries are being sent etc.

RL 95 FAAS DEMO2 PROTO 3 MARQUEE

Have tanker callsign/track/time on textual information of the Marquee.

Implemented.

RL 96 FAAS DEMO2 6.3 B REPLANNING

Error messages, in general should be more descriptive (e.g. "Cannot assign tanker to receiver" should say why).

RL 97 FAAS DEMO2 PROTO 3 SORT

Sort should not distinguish between TOT and station time.

Implemented on the marquee.

RL 98 FAAS DEMO2 PROTO 3 PROCESS

The system should calculate estimated TOT and Station Time based on new estimated Depart time.

This was implemented in ABM. Should there be a "projected" time that is different from a user entered "estimated" time.

RL 99 AAS DEMO2 PROTO 3 STATUS DISPLAY BOARDS

Would like to be able to re-order the columns of the SDBs.

Done. Tailoring.

RL 100 FAAS DEMO2 PROTO 3 TASK ALLOCATOR

Deadline and priority should be optional fields on the Task Allocator window.

Implemented.

RL 101 FAAS DEMO2 PROTO 2 TASK

Each D.O. should have options other than display all Tasks.

4 options are given: Show all tasks, show all active tasks, show my tasks, and

show my active tasks.

RL 102 FAAS DEMO2 PROTO 3 MARQUEE

Put MSN #, Type A/C, and # A/C before every mission symbol on the marquee. This will allow you to display three additional pieces of information in the three columns.

Implemented.

RL 103 FAAS DEMO2 6.3 B MARQUEE

A Replanned mission should show up more on the marquee. Either bold or brighter color or just dont show any of the missions that havent changed.

The current design does not give the information that is needed.

RL 104 FAAS DEMO2 6.3 B MARQUEE

Be able to set time that the Marquee comes up to.

Defaults to the size of a query. - a default time may not be part of the query for 6.3 A.

RL 105 FAAS DEMO2 6.3 B MARQUEE

Would like option of auto-scroll.

The marquee does not move much, even in an hour, this should be a low priority.

RL 106 FAAS DEMO2 PROTO 3 MARQUEE

Combine the number A/C column with the A/C Type column.

Implemented.

RL 107 FAAS DEMO2 PROTO 3 MARQUEE

Marquee should have automatic refresh, i.e. auto-updates.

Implemented.

RL 108 FAAS DEMO2 PROTO 3 MARQUEE

Improve the color schemes/ color contrasts of the marquee.

Implemented.

RL 109 FAAS DEMO2 6.3 B STATUS DISPLAY BOARDS

Have a print selection capability, i.e. don't want to print entire ATO. Also the print function should have an oops button.

Can currently print an SDB, but there is no oops button.

RL 110 FAAS DEMO2 6.3 B TARGET WORKSHEETS

The Target Worksheets should have real time updates.

RL 111 FAAS DEMO2 PROTO 3 ATTACK STATUS REPORT

Do not display system data.

Implemented.

RL 112 FAAS DEMO2 6.3 B LOGIN SCREEN

There should be a theatre set up screen to identify all D.O.s and their responsibilities.

RL 113 FAAS DEMO2 6.3 B USER INTERFACE

Theatre specific operating procedures should be able to be displayed from the UI.

RL 114 FAAS DEMO2 PROTO 3 TASKING

When a task is sent the sender should get an automatic reply when the task is looked at.

Status has a received value.

RL 115 FAAS DEMO2 PROTO 3 TASKING

The Task priority structure should be different from the Target priority.

Urgency used for tasking priorities.

RL 116 FAAS DEMO2 PROTO 3 TASKING

Would like to be able to give other D.O.s a heads up that a task is in progress which may affect them.

Alarms have been implemented.

RL 117 FAAS DEMO2 PROTO 3 TASK NOTIFY

Have the Task Notify Window convey the priority of the task. Also show who the task has arrived from and when it was sent.

Added information to the task notify and subtask returned windows.

RL 118 FAAS DEMO2 PROTO 3 TASK WORKSHEET

Re-arrange the Plans, Attack Replanning, etc buttons in the order that they will be used. Also make it more obvious that they contain pull-down menus.

Implemented putting task button last and moving ATM button to main menu.

RL 119 FAAS DEMO2 PROTO 3 TASKING

Show the SODO if and when a Test has been accomplished on a task.

Tested and Test status added to tasking information.

RL 120 FAAS DEMO2 ATTD TASK WORKSHEET

The process of going from Task familiarization to Task evaluation, to Task implementation should be more obvious.

Hot keys help some, want to show affected missions from task on marquee for ATTD.

RL 121 FAAS DEMO2 PROTO 3 TASK WORKSHEET

Include time sent and change Pending to Working.

Creation time added and working was added as a status.

RL 122 FAAS DEMO2 PROTO 3 MARQUEE

Would like to be able to unassign resources from the Marquee.

Divert, Alert, and Cancel are possible from the Marquee.

RL 123 FAAS DEMO2 6.3 B MARQUEE

The marquee should be able to display late missions, e.g. a 15 minute late mission would turn red.

ABM will have to reason with when a mission is in trouble. (Not a flat time for all missions).

RL 124 FAAS DEMO3 MARQUEE

Cancelled missions should still be displayed on the marquee, perhaps in a subdued color.

RL 125 FAAS DEMO3 MARQUEE/ALERTS

If no status reports have been recieved on a mission currently being flown, (i.e. past the timeline) then the user should be alerted to contact the unit reponsible for the mission. This feature should be user selectable.

RL 126 FAAS DEMO3 ASSIGNING MULTIPLE MISSIONS

The users would like to be able to assign multiple missions at one time, to a tanker or target, etc.

RL 127 FAAS DEMO3 MARQUEE COLORS

White missions on a tan background are hard to see. This combination may be acceptable for cancelled missions, but a different combination should be chosen for the replanning data.

RL 128 FAAS DEMO3 MARQUEE AUTOSCROLL

The marquee should have the user selectable ability to autoscroll. This would be nice for large screen projection.

RL 129 FAAS DEMO3 MARQUEE PRINTING

The marquee needs to be able to be printed. It could be printed in sections and put in a folder.

RL 130 FAAS DEMO3 MARQUEE LEGEND

The marquee legend should have the graphical symbols included and explained.

RL 131 FAAS DEMO3 MARQUEE COLORS

The marquee colors should be user tailorable.

RL 132 FAAS DEMO3 MARQUEE

The marquee and its fonts should be scalable so it could be more readable or so more missions could be displayed than is currently possible.

RL 133 FAAS DEMO3 MARQUEE

The amount of time displayed on the marquee should be tailorable.

RL 134 FAAS DEMO3 MARQUEE

Sorting on the marquee should allow for ascending and descending on each column.

RL 135 FAAS DEMO3 STATUS DISPLAY BOARD PRINTING

When the user is setting up his SDB he needs to know how much of the data will be printed. This should correspond to the specific printer he has selected and its capabilities Printing the SDB must also take into account whether the user wants the extra part of the SDB printed on a seperate sheet which he can the paste together or if he wants two line printing or if he wants to print sideways on a sheet.

RL 136 FAAS DEMO3 STATUS DISPLAY BOARD

The SDB will need to handle aircraft from different services as well as allied aircraft. The SDB also has to be able to handle all mission types, to include airlift, missiles, and perhaps special ops.

RL 137 FAAS DEMO3 STATUS DISPLAY BOARDS

There needs to be a way to query for all packaged missions.

RL 138 FAAS DEMO3 STATUS DISPLAY BOARD QUERIES

There may be a need for more than 5 default query settings.

RL 139 FAAS DEMO3 TASKING

The task notify window's colors and number of beeps should be standardized with CTAPS's SMA module.

RL 140 FAAS DEMO3 TASKING

When a subtask is returned to a DO that DO would like to have the ability to review all the options considered, not just the chosen option.

RL 141 FAAS DEMO3 TASKING

Need to be able to easily log that nothing needs to be done for a task.

RL 142 FAAS DEMO3 TASKING

Self tasking should be made easier, but would be better if it could be eliminated altogether.

RL 143 FAAS DEMO3 TASKING

When a task is being worked the affected missions should automatically come up on the replanning boards.

RL 144 FAAS DEMO3 MAP

The MAP symbology should be standardized with other CTAPS applications.

RL 145 FAAS DEMO3 ATTACK WORKSHEETS

Mission control information needs to be able to be input by the users. (FAC freqs etc) This functionality should be available in the new APS.

RL 146 FAAS DEMO3 OPTION GENERATION

The word "divert" should be changed to "retask" since divert has the conotation that the mission is already air borne.

RL 147 FAAS DEMO3 OPTIONS GENERATION

Option weights should be user tailorable, by position.

RL 148 FAAS DEMO3 OPTIONS GENERATION

There should be a way to compare the different options, perhaps displayed together on the map.

RL 149 FAAS DEMO3 OPTIONS GENERATION

The criteria that are used to generate the options should be included in the users manual.

RL 150 FAAS DEMO3 OPTIONS GENERATION

The user should be able to see what weightings were used to generate the options he is looking at.

RL 151 FAAS DEMO3 OPTIONS GENERATION

Inputs to the "Bag of Tricks": day only aircraft should get a higher weight for a day mission than day/night a/c; negative weight for day a/c assigned to a night mission; weather affects on scl at the target; radius of action around a target (up to 700 km from target); If a "divert" (retask) is necessary then it is better to only change the last legs of the mission; packages need to be deconflicted geographically in Korea.

RL 152 FAAS DEMO3 DATA BASE

Published is a poor word choice because it implies the dissemination of a ATO or CTO to external agencies. A better word would be "Approved".

RL 153 FAAS DEMO3 ALERTS

In CAFMS certain events trigger the CTAPS SMA to send an alert. This functionality should be included in FLEX.

RL 154 FAAS DEMO3 CHANGE OVER BRIEFS

The briefing templates should have fields that are automatically populated with data from the database. Each template would then only have be set up once and when ever it was subsequently opened the data would be retrieved and put into the appropriate location in the change over briefing text.

RL 155 FAAS DEMO3 POPUP OPTION LISTS

The users would like the popup option menus to be called up in a standard way between APS and FLEX.

RL 156 FAAS DEMO3 LAUNCH AUTHORITY

The AOC will need to be able to delegate the authority to launch ground alert aircraft to the appropriate agencies.

RL 157 FAAS DEMO3 STATISTICS

Some additional statistics were suggested:

How the CAS missions were divided among the various ground units.
How many changes were made as a result of wx, mx, battle changes, etc.

RL 158 FAAS DEMO3 STATUS REPORTS

FLEX will need to be able to get the data from the units once CAFMS goes away. (e.g. take off times, log data{fuels, munitions}, etc.

RL 159 FAAS DEMO3 DEFENSIVE CAPABILITIES

CAFMS defensive capabilities need to be included in FLEX. It was suggested that we talk to some 17XX's to see what the requirements would be.

RL 160 FAAS DEMO3 FLEX REMOTES

FLEX remotes, if any, should allow the units to continue to input data when comm lines go down and then allow that data to be updated when the comm links come back up.

RL 161 FAAS DEMO3 FLEX REMOTES

FLEX remotes, if any, need to be able to be configured with the appropriate read/write authorizations for the unit at which they are installed.

RL 162 FAAS DEMO3 ATO EXECUTION

FLEX needs to be able to monitor and task surface to surface systems as well as any other systems that are on the ATO.

RL 163 FAAS DEMO3 DIVERTED ASSETS

A/C should not have to land at the same base they launched from. FLEX needs to be able to track A/C that land at a forward base or any diverted A/Cs.

RL 164 FAAS DEMO3 CALLSIGNS AND IFF/SIFS

FLEX must know what call signs, mission IDs, and IFF/SIFs it can assign to missions it creates during a specific ATO period.

RL 165 FAAS DEMO3 DUTY OFFICER LOG

The users would like to be able to sort and query on the DO Logs.

RL 166 FAAS DEMO3 ATO CONFS

FLEX needs to save all the ATO CONFs generated throughout the day.

RL 167 FAAS DEMO3 ATO START TIME

FLEX needs to be able to handle an H hour if it is going to execute off the shelf ATOs.

RL 168 FAAS DEMO3 ALERTS

If the CTOs are not going to be disseminated immediately, then there needs to be a way to alert the units of an upcoming change so they can start to plan.

RL 169 FAAS DEMO3 IMPORT

FLEX needs to be able to read in an USMTF compliant ATO, and be able to execute it.

RL 170 FAAS DEMO3 IMPORT

FLEX needs to be able to recieve the latest ACO from ADS.

RL 171 FAAS DEMO3 DATE TIME STAMPS

FLEX produced messages should automatically be assigned a date-time group.

RL 172 FAAS DEMO3 SETUP

The authority to approve and publish tasks should be user tailorable.

RL 173 FAAS DEMO3 ATO DAY

When multiple ATOs are being monitored there needs to be a way to tell which ATO the mission is associated with.

RL 174 FAAS DEMO3 ATO CONFS

When a change is made to the Remarks sections of the ATO, only the changes should be sent out in the ATO CONF and not the entire Remarks sections.

Appendix J:

FLEX Experiment
Data Collection Forms

Figure J-1: General Information for Participants

This study is being conducted to investigate the effects of computer-based systems on the performance of re-planning decision tasks.

If you agree to participate, you will need to attend two half-day sessions. You will be provided basic training in the decision task and the computer system you will use for the session.

This study does not involve any physical, psychological, social, or legal risks. In addition, there is no discomfort involved in the tasks. There are no costs to you or any other party.

Participation in this study is voluntary; you may withdraw from the study at any time for any reason. There is no penalty for not participating or withdrawing. Your participation will allow you to provide input into the development of systems that will support the operational Air Force of the future.

All data collected in this study will be kept confidential; all person-identifiable data will be coded so that you cannot be identified. There will a brief, informal discussion at the close of the session. This discussion will be audio taped to assist in later analysis. All discussions will be kept confidential.

This study is being conducted by Lee Scott Ehrhart from the C^{3}I Center at George Mason University in collaboration with Rome Laboratory and Drexel University. She may be reached at (703) 993-1503 for questions or complaints. You may also contact the George Mason University Office of Research at (703) 993-2295 if you have any questions or comments regarding your rights as a participant in this research.

This project has been reviewed according to George Mason University procedures governing your participation in this research.

Subject #: ___________

Figure J-2: Background Information

Name & Rank: ______________________________ DOB: __________ DOR: __________

Current Assignment : ______________________________ Position : _______________

Time in this Position: ______________________________

Previous Assignments & Positions:	Approximate Period:
______________________________	______________
______________________________	______________
______________________________	______________
______________________________	______________

Did you participate in any operations conducted in the following regions:

Southwest Asia (Persian Gulf)

____ No ____ Yes (please describe) ______________________________

____

____

Baltic States (formerly Yugoslavia)

____ No ____ Yes (please describe) ______________________________

____

____

Northern Africa (Somalia)

____ No ____ Yes (please describe) ______________________________

____

____

List any other experience relevant to tanker operations (exercises, etc.):

____

____

Subject #: ________ Interface: ______ Session Date: __________ AM/PM
Trial #: _______

Figure J-3: Workload Ratings

Instructions: Place a mark on each scale that represents the magnitude of each factor in the task you just performed.

Demands	Ratings for Task
Mental Demand	Low │ ------------------------------------ │ High
Temporal Demand	Low │ ------------------------------------ │ High
Own Performance	Low │ ------------------------------------ │ High
Frustration	Low │ ------------------------------------ │ High
Effort	Low │ ------------------------------------ │ High

Mental Demand:
How much mental and perceptual activity was required (e.g., thinking, deciding, calculating, remembering, looking, searching, etc.)? Was the task easy or demanding, simple or complex, exacting or forgiving?

Temporal Demand:
How much time pressure did you feel due to the rate or pace at which the task had to be performed? Was the pace slow and leisurely or rapid and frantic?

Own Performance:
How successful do you think you were in accomplishing the goals of the task? How satisfied were you with your performance in accomplishing this goal?

Frustration:
How insecure, discouraged, irritated, stressed and annoyed versus secure, gratified, content, relaxed and complacent did you feel during the task?

Effort:
How hard did you have to work (mentally & physically) to accomplish your level of performance?

Subject #: ________ Interface: ______ Session Date: __________ AM/PM

Figure J-4: Workload Comparison Ratings

Instructions: Select the member of each pair that provided the most significant source of workload variation in these tasks.

Temporal Demand / Frustration

Temporal Demand / Effort

Own Performance / Frustration

Own Performance / Effort

Temporal Demand / Own Performance

Temporal Demand / Mental Demand

Own Performance / Mental Demand

Frustration / Mental Demand

Effort / Mental Demand

Effort / Frustration

Mental Demand:

How much mental and perceptual activity was required (e.g., thinking, deciding, calculating, remembering, looking, searching, etc.)? Was the task easy or demanding, simple or complex, exacting or forgiving?

Temporal Demand:

How much time pressure did you feel due to the rate or pace at which the task had to be performed? Was the pace slow and leisurely or rapid and frantic?

Own Performance:

How successful do you think you were in accomplishing the goals of the task? How satisfied were you with your performance in accomplishing this goal?

Frustration:

How insecure, discouraged, irritated, stressed and annoyed versus secure, gratified, content, relaxed and complacent did you feel during the task?

Effort:

How hard did you have to work (mentally & physically) to accomplish your level of performance?

Subject #: ________ Interface: ______ Session Date: ___________ AM/PM

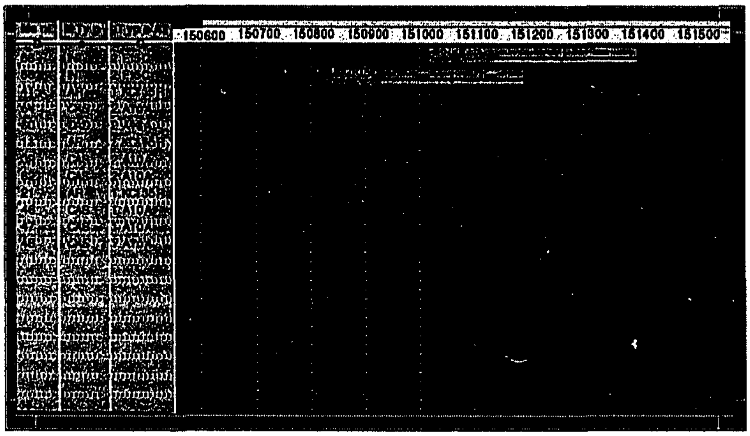
Figure J-5: Interface Evaluation

1. What did you like and/or find most helpful about this interface?

2. What did you dislike and/or find most difficult about this interface?

Subject #: ________ Interface: ______ Session Date: __________ AM/PM

Figure J-6: Subjective Evaluation



Option View

1. **Problem Identification.**
   To what extent did this window contribute to your identification of the problem (i.e., the task)?

   not at all **somewhat** **greatly**

   1 2 3 4 5 6 7 8 9 10 11

2. **Situation Assessment.**
   To what extent did this window contribute to your understanding of the situation (i.e., location & scheduled availability of resources)?

   not at all **somewhat** **greatly**

   1 2 3 4 5 6 7 8 9 10 11

3. **Option Evaluation.**
   To what extent did this window contribute to your understanding and evaluation of the options presented?

   not at all **somewhat** **greatly**

   1 2 3 4 5 6 7 8 9 10 11

4. **Option Selection.**
   To what extent did this window contribute to your selection of the better option?

   not at all **somewhat** **greatly**

   1 2 3 4 5 6 7 8 9 10 11