

AD-A128 916

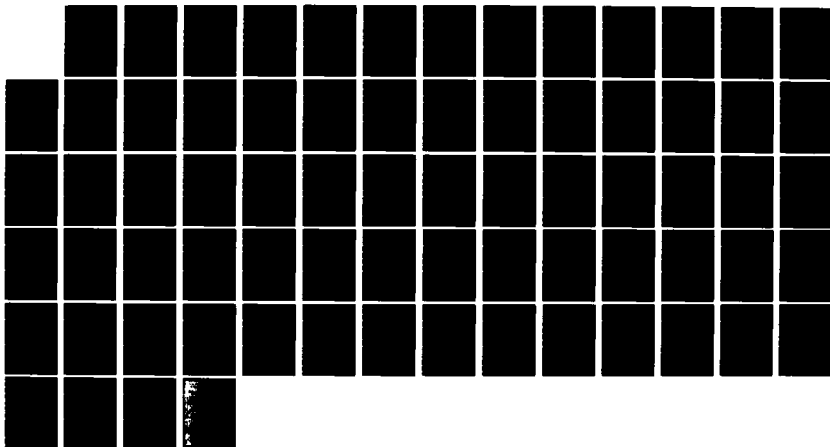
SOFTWARE TECHNOLOGY FOR ADAPTABLE RELIABLE SYSTEMS
(STARS) FUNCTIONAL TAs. (U) OFFICE OF THE DEPUTY UNDER
SECRETARY OF DEFENSE RESEARCH ANDEE. 30 MAR 83

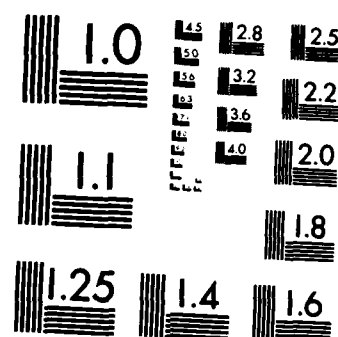
1/1

UNCLASSIFIED

F/G 9/2

NL





MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM	
1. REPORT NUMBER	12. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER	
4. TITLE (and Subtitle) Software Technology For Adaptable Reliable Systems (STARS) Functional Task Area Strategy For Human Engineering		5. TYPE OF REPORT & PERIOD COVERED	
7. AUTHOR(s) The DoD Joint Service Task Force on the Software Initiative (STARS)		6. PERFORMING ORG. REPORT NUMBER	
PERFORMING ORGANIZATION NAME AND ADDRESS		8. CONTRACT OR GRANT NUMBER(s)	
10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS			
11. CONTROLLING OFFICE NAME AND ADDRESS Deputy Under Secretary of Defense Research & Advanced Technology Washington, D.C. 20301		12. REPORT DATE March 30, 1983	
13. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)		13. NUMBER OF PAGES	
		14. SECURITY CLASS. (of this report) Unclassified	
		15. DECLASSIFICATION/DOWNGRADING SCHEDULE	

DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited

DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

Unclassified

16. SUPPLEMENTARY NOTES

17. KEY WORDS (Continue on reverse side if necessary and identify by block number)

STARS, software initiative, Ada, hardware, automated tools, support environments, functional task areas, life-cycle costs, embedded computers, mission critical, military systems, adaptable, reliable, system interface standards, software engineering institute, deliverables, application specific environment, linkage, coordination.

ABSTRACT (Continue on reverse side if necessary and identify by block number).

This document identifies the scope, sub-objectives and strategies designed to provide the conceptual approach for accomplishment of the STARS Program objectives in the human engineering functional task area. Its main purpose is to help guide the implementation planning process.

DTIC FILE COPY

FORM 1473
1 JAN 79EDITION OF 1 NOV 65 IS OBSOLETE
S-N 8182-LF-814-8681

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

**SOFTWARE TECHNOLOGY FOR
ADAPTABLE, RELIABLE SYSTEMS (STARS)
FUNCTIONAL TASK AREA STRATEGY FOR
HUMAN ENGINEERING**



Department of Defense

30 March 1983



Accession For	
NTIS QUART	
DTIC	
IMP	
2	
A	

83 06 01 055

FOREWORD

This strategy document is one of eight functional task area strategies produced by the STARS Joint Task Force. All of the documents produced by the Task Force, including the general STARS Program Strategy document, are listed in the STARS Joint Task Force Report.

This document identifies the scope, sub-objectives and strategies designed to provide the conceptual approach for accomplishment of the STARS Program objectives in the human engineering functional task area. It identifies and describes the high-level activities, products and capabilities. In order to provide full understanding, background and rationale material is sometimes covered that is also in STARS Program Strategy.

These functional task area strategy documents do not attempt to delineate the detailed plans, costs and procedures for bringing the proposed products and capabilities into being and do not identify the form of the particular projects that will undertake the work nor the organizations in which the work will be accomplished. Instead, these strategies are intended to guide the process of such implementation planning and accomplishment.

Indeed, because of the high degree of linkage among the functional task areas, implementation plans and acquisitions may well combine related capabilities and products across areas. Individual projects may tackle only part of one subtask from a functional area or several subtasks from several functional areas.

Thus, this functional task area strategy describes broad, achievable requirements for accomplishing the relevant STARS objectives. Its main purpose is to help guide the implementation planning process.

Ada[®] is a Registered Trademark of the Department of the Defense,
Ada Joint Program Office.

ACKNOWLEDGEMENTS

In addition to the expertise provided by the panel and by the workshop attendees, this plan has benefited from the contributions of the following people:

John Bailey	- General Electric Company
Deborah Boehm-Davis	- General Electric Company
Larry Druffel	- Computer Software and Systems, OUSD(R&AT)
Marlene Hazle	- MITRE
Tom Love	- Schlumberger-Doll Research
Rudy Ramsey	- ITT
Sam Redwine	- MITRE
Bill Riddle	- Software Design & Analysis
Ben Shneiderman	- University of Maryland
Tony Wasserman	- University of California

TABLE OF CONTENTS

	<u>Page</u>
1.0 OVERVIEW	1
1.1 Scope of the Task Area	1
1.2 Strategy	2
2.0 MAJOR SUBTASKS AND DETAILED ACTIVITIES	4
2.1 Subtask 1: Methodology for Incorporating Human Engineering	4
2.1.1 Rationale	4
2.1.2 Premises	6
2.1.3 Description	8
2.1.4 Coordination	9
2.1.5 Deliverables	9
2.1.6 References to Milestone Charts	10
2.2 Description of Detailed Activities for Subtask 1	10
2.2.1 Detailed Activity 1.1: Embedded Systems Survey	10
2.2.1.1 Purpose and Description	10
2.2.1.2 Coordination	11
2.2.1.3 Deliverable	11
2.2.2 Detailed Activity 1.2: Existing Practices Survey	11
2.2.2.1 Purpose	11
2.2.2.2 Description	11
2.2.2.3 Deliverables	11
2.2.3 Detailed Activity 1.3: Focusing Mechanism	12
2.2.3.1 Purpose	12
2.2.3.2 Description and Coordination	12
2.2.3.3 Deliverables	13
2.2.3.4 Cost Factors	13
2.2.4 Detailed Activity 1.4: Development of Methodology	14
2.2.4.1 Rationale	14
2.2.4.2 Premises	14
2.2.4.3 Description	15
2.2.4.4 Coordination	16
2.2.4.5 Deliverables	16
2.2.5 Detailed Activity 1.5: Design Guidelines	17
2.2.5.1 Purpose	17
2.2.5.2 Premises	17
2.2.5.3 Description	18
2.2.5.4 Coordination	18
2.2.5.5 Deliverables	19
2.2.6 Detailed Activity 1.6: Supporting Tool Development	19
2.2.6.1 Rationale	19

TABLE OF CONTENTS (Continued)

	<u>Page</u>
2.2.6.2 Inputs	19
2.2.6.3 Description	20
2.2.6.4 Coordination	21
2.2.6.5 Deliverables	21
2.2.7 Detailed Activity 1.7: Methodology and Tools Applied	21
2.2.7.1 Rationale	21
2.2.7.2 Inputs and Premises	22
2.2.7.3 Description	22
2.2.7.4 Coordination	22
2.2.7.5 Deliverables	22
2.2.8 Detailed Activity 1.8: Methodology and Tools Enhanced	23
2.2.8.1 Purpose	23
2.2.8.2 Description	23
2.2.8.3 Coordination	23
2.2.8.4 Deliverables	23
2.3 Human Engineering of the Support Environment	24
2.3.1 Purpose	24
2.3.2 Inputs	25
2.3.3 Description	25
2.3.4 Coordination	25
2.3.5 Deliverables	26
2.3.6 References to Milestone Charts	26
2.4 Description of Detailed Activities for Subtask 2	26
2.4.1 Detailed Activity 2.1: Prototype Workstation	26
2.4.1.1 Purpose	26
2.4.1.2 Premises and Inputs	26
2.4.1.3 Description	27
2.4.1.4 Coordination	27
2.4.1.5 Deliverables	28
2.4.2 Detailed Activity 2.2: Support Environment Interface	28
2.4.2.1 Purpose	28
2.4.2.2 Premises	29
2.4.2.3 Description	29
2.4.2.4 Coordination	30
2.4.2.5 Deliverables	30
2.5 Subtask 3: Research Program in Human Engineering	31
2.5.1 Purpose	31
2.5.2 Description	32
2.5.3 Coordination	32
2.5.4 Deliverables	32
2.5.5 References to Milestone Charts	33
2.6 Detailed Descriptions of Activities for Subtask 3	33

TABLE OF CONTENTS (Continued)

	<u>Page</u>
2.6.1 Detailed Activity 3.1: Research Review	33
2.6.1.1 Purpose	33
2.6.1.2 Inputs and Premises	33
2.6.1.3 Description	33
2.6.1.4 Coordination	33
2.6.1.5 Deliverables	33
2.6.2 Detailed Activity 3.2: Research	34
2.6.2.1 Rationale	34
2.6.2.2 Inputs	34
2.6.2.3 Description	34
2.6.2.4 Coordination	36
2.6.2.5 Deliverables	37
2.7 Subtask 4: Evaluation of Human Engineering Impact	37
2.7.1 Purpose	37
2.7.2 Inputs	37
2.7.3 Coordination	38
2.7.4 Deliverables	38
2.7.5 Reference to Milestone Chart	39
2.8 Description of Detailed Activities for Subtask 4	39
2.8.1 Detailed Activity 4.1: Human Engineering Goals	39
2.8.1.1 Purpose	39
2.8.1.2 Inputs and Premises	40
2.8.1.3 Description	40
2.8.1.4 Coordination	40
2.8.1.5 Deliverables	40
2.8.2 Detailed Activity 4.2: Measures and Feasibility	41
2.8.2.1 Purpose	41
2.8.2.2 Inputs	41
2.8.2.3 Description	41
2.8.2.4 Coordination	42
2.8.2.5 Deliverables	42
2.8.3 Detailed Activity 4.3: Data-Collection Requirements	42
2.8.3.1 Purpose	42
2.8.3.2 Inputs and Premises	42
2.8.3.3 Description	43
2.8.3.4 Coordination	43
2.8.3.5 Deliverables	44
2.8.4 Detailed Activity 4.4: Analysis and Interpretation	44
2.8.4.1 Purpose	44
2.8.4.2 Inputs	44
2.8.4.3 Description	44
2.8.4.4 Coordination	44

TABLE OF CONTENTS (Continued)

	<u>Page</u>
2.8.4.5 Deliverables	45
3.0 ANALYSIS OF BENEFITS	46
4.0 REFERENCES CITED	48
5.0 BOOKS	50
5.1 Journals	50
5.2 Books	50
5.3 Workshops	52
6.0 LINKAGES TO OTHER TASK AREAS	53
6.1 Linkage to Human Resources	53
6.2 Linkage to Project Management	54
6.3 Linkage to Support Systems	55
6.4 Linkage to Application Specific	56
6.5 Linkage to Technology Insertion	57
6.6 Linkage to Measurement	57
6.7 Linkage to Acquisition	57
6.8 Linkage to Systems	58
APPENDIX I HISTORY OF THE HUMAN ENGINEERING PLAN	59

1.0 OVERVIEW

1.1 Scope of the Task Area

The STARS effort has been initiated to provide the technology and organizational mechanisms to meet the demands for increased software productivity, functionality, and system reliability imposed by DoD mission requirements. As part of this effort, a joint service task force was established to analyze the problems in existing software development technologies and mechanisms. The summary of their analysis, "Report of the DoD Joint Service Task Force on Software Problems, 30 July 1982," identifies human engineering as a significant problem in all phases of the software life cycle for embedded computer systems. Insufficient attention to human engineering is evident in the support systems for development teams of mission-critical, software-intensive systems. In addition, the interfaces between the end user and these systems are also poorly engineered. This not only creates an impediment to productivity (for both the software developer and the end user), but also creates potentially dangerous situations. The end user of real-time, mission-critical systems, such as weapons or avionics systems, cannot tolerate an unfriendly or balky computer.

Since the 1940's, the discipline of human engineering has successfully applied knowledge of human capabilities and limitations to the design of military systems to achieve optimal user effectiveness, efficiency, comfort, and safety compatible with system requirements. The Human Engineering Task Area should utilize that knowledge base concerning users' physical and cognitive limitations and extend it to the arena of computer hardware and software to produce methodologies and tools which will enhance productivity and quality throughout the life cycle of embedded computer systems. While this task area will focus on the user in the broadest sense, attention should be primarily directed to two major groups of users: the end users of DoD

embedded computer systems, and the personnel involved in software development and support. For this latter group, the Human Engineering Task Area is concerned not only with the user interface to the automated support environment but with the human factors of the entire software development and support process.

1.2 Strategy

For the short term, significant benefits can result from the systematic application of current knowledge to develop methodologies and tools. For the longer term, advances could rest on an increased understanding of how individuals and teams interact with computers and with each other to complete their required tasks. This understanding will accelerate the growth and definition of a true engineering discipline in which basic human-computer interaction principles would be developed, verified, and used. This plan capitalizes on the current state of the art while initiating a program of research and systematic experimentation to support further advances. The plan can accomplish this through four major subtasks which consist of the following:

- 1) development and continual enhancement of a general human engineering methodology (or methodologies) along with supporting tools,
- 2) design of the user interface for the automated support environment,
- 3) experimentation to establish and test basic principles and models of human-computer interaction and problem solving, and
- 4) identification of measures to assess progress.

The general methodology (Subtask 1) would provide a set of procedures, guidelines, and supporting tools for incorporating human engineering principles into each phase of the development cycle of a software system. The development of this methodology should build on

past and current efforts whenever possible. It should also involve a substantial amount of further research and development. Both the guidelines and the methodology should be verified as a result of experience on actual projects and refined as a result of increased knowledge of the principles underlying human-computer interaction.

As noted above, the Human Engineering Task Area should be concerned not only with the end user of application software but with the user of automated support environments as well. This latter group includes the management, technical, and staff personnel responsible for software development and support. Human Engineering should play a major role in the design of prototype workstations as well as in the design of the user interface to all tools, system functions, on-line help facilities, etc. (Subtask 2).

Further advances in human engineering should rest on the construction and testing of models of human problem solving and human-computer interaction and on controlled experimentation to verify principles and guidelines (Subtask 3).

The Human Engineering Plan is intended to have a major impact on the usability of future DoD computer-based systems; these include systems for the software developer (i.e., automated support environments) as well as end-product embedded systems. The Human Engineering Plan is also intended to have a major impact on the entire process of software development and support. The identification of measures (Subtask 4) would be essential for establishing a baseline and assessing this impact.

Each of the four subtasks are described in the next section of this plan (Section II).

2.0 MAJOR SUBTASKS AND DETAILED ACTIVITIES

2.1 Subtask 1: Methodology for Incorporating Human Engineering

2.1.1 Rationale

A primary objective of the Human Engineering Task Area is to incorporate human engineering principles into the design of all computer-based systems that interface with a human user. The characteristics that define good human engineering - such as ease of learning, flexibility and efficiency - cannot be added on at the end of system development but must be an integral part of the design from the beginning. There are a number of activities which must be performed during each development phase (i.e. requirements analysis, system specification, etc.) to insure this. What is needed is a methodology that focuses on human factors issues at all stages of the system development process.

Ramsey and Atwood (1980) came to a similar conclusion following an extensive literature search to assess the state of the art in the human engineering of computer systems (sponsored by the Office of Naval Research). They suggested that what is needed is a "design guide" which discusses the major human engineering issues arising throughout the development phase. Smith (1982) has also pointed to the need for an integrated human engineering methodology.

As part of a research effort sponsored by the Office of Naval Research, Johnson and Hartson (1982) have taken this idea one step further and have argued for a "dialogue author" who is a specialist in communication and in human factors and who is distinct from the traditional designer or programmer. Johnson and Hartson suggest that there is a need to establish and maintain a strict independence between interface-handling components of a system and computational components. The primary advantage of this independence is that deci-

sions relating to the dialogue are not "hard wired" into the rest of the system. Instead, the dialogue designer can experiment with and improve the interface during development as issues and alternatives become clearer. Johnson and Hartson use the term "dialogue management" to refer to an emerging discipline dealing with "...the creation, modification, simulation, execution, testing, and metering of dialogues in an integrated manner." This notion of dialogue management is similar to the idea of a human engineering methodology. The idea of a dialogue author who is distinct from the traditional designer or programmer is one of a number of potentially useful ideas deserving of further investigation.

The consistent application of the methodology will be greatly facilitated by the use of an appropriate set of tools. These include automated tools as well as non-automated tools or aids such as design guidelines. Tools will be needed for all phases of the development cycle, both to facilitate or enforce adherence to the methodology and to ease the transition between phases. The development of tools to support the methodology is included as an activity within this subtask.

One such tool is represented by design guidelines for the user-system interface. There are currently several ongoing efforts to compile design guidelines. Guidelines provide one means of summarizing current knowledge and judgment for use by interface designers. Another activity within this subtask is the continuation and extension of these efforts.

Much can be learned from attempts to apply the methodology and tools prior to recommending their widespread use. This experience will provide useful feedback by uncovering deficiencies and gaps. In addition, the selected projects can serve as realistic models for later projects. The application of the methodology and tools is an additional activity within this subtask.

The effort to develop an effective methodology is a continuing one. The methodology should not be frozen at the end of its development but should continue to be enhanced and refined. This is also included as an activity within the subtask. It is important to emphasize that the human factors methodologies must complement and be integrated with the other methodologies involved in the development of the end product system. They must, of course, operate within the functionality, reliability, budgeting and scheduling constraints of the project.

The human engineering literature is widely scattered across several disciplines which include computer science, psychology, human factors, ergonomics, and industrial design. Because of the interdisciplinary nature of this area, there is a particularly critical need to focus and coordinate further efforts while, at the same time, receiving inputs from the relevant disciplines. One way to accomplish this is through an Advisory Panel consisting of a small group (perhaps 7 or 8) of leading practitioners in the relevant related disciplines. This group should be responsible for recommending and reviewing high priority areas for methodology and tool development and for recommending and reviewing research efforts in human engineering. It would serve as a support group for the DoD organization responsible for carrying out the Human Engineering Plan. It would closely track the efforts of all four subtasks throughout the span of the Software Initiative and would continually re-assess priorities and recommend funding for areas with the highest payoff.

2.1.2 Premises

A question arises concerning the extent to which a single human engineering methodology will suffice for all relevant applications and for all groups of users. For example, will the same set of procedures apply to the development of the user interface for a programming support environment as compared to a flight control system? It

is assumed that there is a sufficiently high degree of similarity in the basic types of activities which are needed to ensure good human engineering that a single, general methodology can be defined across different applications and user groups. There may, however, be sufficient variation to more properly refer to a class of related methodologies rather than to a single methodology. It is also assumed that a common set of principles will apply across applications. The specific design goals, however, may vary.

Consider, for example, the differences in design goals as a function of the characteristics of the users - in particular, their computer background and frequency of computer usage. Shneiderman (1983) distinguishes between three basic groups of users as follows:

- 1) computer-naive, relatively infrequent users,
- 2) computer-naive, frequent users, and
- 3) computer specialists.

Shneiderman points out that ease of learning and retention are important design goals for the computer-naive, infrequent user; efficiency is generally of lesser importance. For the computer naive but frequent user, ease of learning is initially important but is soon replaced by concerns for efficiency and flexibility; multiple interfaces may be especially important for this group. Finally, efficiency and flexibility should be important goals in designing systems for the computer specialist.

In addition to differences in the design goals as a function of the users and application, there may be some variation in the specific activities which are undertaken in designing and evaluating the user interface. For example, in some cases extensive experimentation with prototypes may be required, particularly for first-time users. In other cases, a paper and pencil form of evaluation based on an appropriate analytic model may be sufficient (Reisner, 1982).

The important point is that some type of evaluation be conducted prior to actual implementation. A major purpose of the methodology should be to force attention to these types of issues. (It is, of course, also the case that different procedures and tools for evaluation will come into play at different points in the development cycle.)

There is a need for a survey of DoD embedded systems applications to determine the relative amount of user-system interaction within different applications and the general nature of that interaction. This survey would help in setting the priorities of the Human Engineering Task Area so that effort is focused on life-critical systems with a high degree of user-system interaction.

2.1.3 Description

A survey of DoD embedded applications should be conducted to establish a baseline for setting priorities.

A survey of existing tools and practices related to the human engineering of user interfaces would be conducted. The activities underlying the development of the methodology (as well as those underlying the other Human Engineering subtasks) would be integrated and focused through establishment of an interdisciplinary Advisory Panel.

This subtask would involve the development of a general and integrated methodology for incorporating human engineering into system development. The methodology will point to relevant activities during each phase including requirements analysis, functional specifications, design, implementation, and operations.

This subtask would also involve the development of an integrated, extensible set of tools for designing, implementing, and evaluating the user interface. This would include the development

and continual enhancement of a comprehensive set of design guidelines.

In addition, this subtask would involve the application of the methodology to selected projects to obtain feedback on areas of success and difficulty.

The methodology and tools would be continually refined and enhanced as a result of experience on actual projects and the discovery of new principles of human-computer interaction. In fact, it is important that the methodology be extensible - that it be capable of incorporating new and perhaps radically different principles of interface design.

2.1.4 Coordination

It is essential that the developers of the human engineering methodology be closely coordinated with the Support Systems Task Area which is responsible for integrating methods and tools. The human engineering methodology should consist of a set of procedures and work products which parallel (and which should not interfere with) other system development activities. The procedures developed for human engineering as well as the supporting tools must be integrated into the more general development methodology.

There should also be coordination with the Measurement Task Area to assist in identifying points in the system development cycle for data collection and other forms of quantitative assessment of user-system interfaces.

2.1.5 Deliverables

There are eight major sets of deliverables under this subtask:

- 1) a survey of DoD embedded applications, focusing on the extent and nature of the end-user interaction,

- 2) a survey of existing human engineering practices, procedures, and tools,
- 3) specific suggestions by the Advisory Panel identifying key areas of the methodology (procedures, tools, work products, etc.) for priority funding,
- 4) an integrated set of procedures for incorporating human engineering into system development (i.e., the methodology),
- 5) a comprehensive set of design guidelines for user-system interfaces,
- 6) other tools in addition to design guidelines which will form an extensible, integrated set,
- 7) results of applying the methodology to actual projects with recommendations for refinements and enhancements, and
- 8) periodic enhancements to the methodology and tool set.

2.1.6 References to Milestone Charts

The major subtasks are shown in Figure 1. The detailed activities for this subtask are shown in Figure 2.

2.2 Description of Detailed Activities for Subtask 1

2.2.1 Detailed Activity 1.1: Embedded Systems Survey

2.2.1.1 Purpose and Description. The purpose of this survey is to identify the major human factors issues within different applications of DoD embedded systems. For each application, the following information should be obtained:

- extent of user-system interaction
- nature of the interaction (e.g., mission critical?)
- characteristics of the hardware and software interface
- extent to which human factors issues are typically addressed during system development

- areas most in need of improvement

This survey would establish a baseline for the identification of critical and/or high payoff applications for the human engineering of embedded systems.

2.2.1.2 Coordination. This activity should be reviewed by the Application-Specific Task Area which can provide useful input. The survey would be turned over to the Advisory Panel. It should help insure that the human engineering activities are responsive to the needs of the end users of DoD embedded systems.

2.2.1.3 Deliverable. The deliverable for this activity would be a survey of DoD embedded systems, focusing on the extent and nature of the end-user interaction for each major application.

2.2.2 Detailed Activity 1.2: Existing Practices Survey

2.2.2.1 Purpose. This survey of existing human factors procedures and tools should provide a starting point for the development of a coherent methodology. As much as possible, efforts would be made to utilize existing tools and to learn from the successes and failures of others.

2.2.2.2 Description. This activity would involve a survey of current tools and procedures for human engineering of user interfaces. This survey would be used in the evaluation to establish priorities for integration and focus on specific tools and methodologies.

2.2.2.3 Deliverables

The deliverable for this activity would be a report describing currently available tools for the design, implementation, and evaluation of user interfaces. The survey would also outline specific procedures followed for incorporating human factors into interface design.

2.2.3 Detailed Activity 1.3: Focusing Mechanism

2.2.3.1 Purpose. The success of this functional task area strategy would be largely determined by the extent to which the various detailed activities address the appropriate issues and problems and by the extent to which they are integrated and focused on a common set of goals. There was a consensus among the panel and attendees at the February STARS workshop that the general issues addressed by this plan are, in fact, the appropriate ones. However, there was also a consensus that there is a need for further discussion and deliberation.

The identification of specific high priority activities and tools and of specific research projects requires further planning and continual feedback and re-assessment. The human engineering of computer systems is currently an inter-disciplinary area. There is no single, well-established discipline with a prior history and a clear path for future progress. There is a growing interest in this important area and a number of promising but scattered research, development, and application efforts. The Human Engineering activities are likely to reflect this state of affairs unless a specific mechanism is put into place to insure a focused, effective effort. The mechanism suggested is an Advisory Panel for the purpose of evaluation of tools and methodologies.

2.2.3.2 Description and Coordination. The Advisory Panel would consist of a small group (7 or 8) of leading practitioners and researchers from the relevant disciplines (e.g., cognitive psychology, software engineering, human factors). They would be selected during Year 0 by the DoD organization responsible for implementing the Human Engineering Plan. They would serve that organization in an advisory (and not a supervisory) role. Key areas for methodology and tool development as well as for further research would be identified.

(Research activities are part of Subtask 3.) The Advisory Panel provides a focus for the Human Engineering Task Area while insuring that multiple disciplines are represented. The Advisory Panel should also serve as a liaison between the various activities within the Task Area to insure that they are mutually responsive to the needs of the Area and to the STARS program as a whole.

The Advisory Panel would use the survey of DoD embedded systems as one major source of input and the survey of existing human engineering practices and tools as another. It would also be given a review and assessment of current and past research. (The latter would be produced as part of Subtask 3.)

2.2.3.3 Deliverables. The deliverables would consist of specific suggestions for methodology and tool development as well as for further research. These suggestions would form the basis for RFP's. It is important that other, potentially useful ideas not be excluded by this process. The Advisory Panel would also provide periodic assessments of progress in the Human Engineering Task Area and continually updated suggestions for further work.

2.2.3.4 Cost Factors. This activity is estimated to require 50% time for seven people, 100% time for one person (the Panel chair) plus travel funds for all members. The Panel should extend over the entire span of the STARS program.

Half time for an Advisory Panel seems very high. The Air Force has organizations like the Scientific Advisory Board and the Air Force Studies Board who are experts in their fields, and most of the labor is voluntary -- for the prestige. The cost estimate appears to be valid since this is to be a working panel, unlike other such panels. Further, this detailed activity has been somewhat revised to focus on its product (priorities for tools, methodologies, research, and experimentation) vice the mechanism, which is the Panel.

2.2.4 Detailed Activity 1.4: Development of Methodology

2.2.4.1 Rationale. As noted earlier, the desired characteristics of a user interface will not emerge automatically. They must be made explicit and designed-in from the beginning. In fact, human engineering principles must be incorporated during the early stages of requirements analysis and continue into field operation. What is needed is a general, integrated set of activities and procedures for guiding the design and development of the user interface along. The systematic and widespread application of these procedures would mark the emergence of human engineering as a true discipline.

For the methodology to be effective, it is essential that it be integrated, that is, that earlier activities be directly related to later activities. An example of this integration can be seen in Smith's (1982) efforts to develop a checklist for identifying the functional capabilities required of the user interface. This checklist is organized around the same functional areas (data entry, data display, etc.) as his design guidelines, thus easing the transition between an identification of needed capabilities and consideration of the relevant guidelines.

2.2.4.2 Premises. As noted earlier, Hartson and Johnson (1982) have argued for the necessity of a "dialogue author" who is distinct from the traditional designer or programmer. There are two major advantages of this approach:

- 1) It encourages the development of a specialist who is familiar with the issues of interface design.
- 2) It encourages the design of a conceptually integrated interface and, conversely, it avoids the problems associated with inconsistencies in the interface as a consequence of different design and implementation decisions made by different individuals, often as an afterthought to the design.

This is an idea that should be investigated further.

2.2.4.3 Description. This activity should involve the development of a general, integrated methodology for incorporating human engineering principles into the system development process. The methodology would build on existing work and would describe relevant activities during each development phase. It would also include a discussion of the major issues which arise in each phase. The following list contains examples of relevant activities. It makes no claim for completeness.

Requirements Analysis:

- analyzing users and their activities, goals, terminology
- determining the relevant user characteristics which have consequences for interface design
- establishing quantitative acceptance tests of system usability (Shneiderman, 1983)

Functional Specifications:

- identification of the functional capabilities required

Design:

- application of design guidelines and other design tools
- documentation aids for the user interface (for the designers and for the users)

Evaluation Techniques:

- use of prototypes and simulations combined with empirical data collection
- use of tools to check for adherence to guidelines

Implementation:

- insuring consistency with the design

Operational Use:

- monitoring the profile of system usage in the field

In addition to developing procedures and discussing relevant issues, this activity would result in suggestions concerning supporting tools. As an example, if the use of prototypes of the user interface early in design is advocated, a useful tool would be a display/dialogue generator which allows for the rapid construction of these prototypes.

2.2.4.4 Coordination. It is essential that the development of the guidelines (Detailed Activity 1.5) and other tools (Detailed Activity 1.6) be closely coordinated with the development of the methodology.

There should also be a close interaction between the developers of the methodology and the Support Systems Task Area. As mentioned earlier, the human engineering methodology must be consistent with other development methodologies and must operate within the budget, schedule, and other constraints of any given project. In addition, attempts to incorporate human engineering principles into the design of the user interface for the automated support environment (Subtask 2) should be closely monitored by the developers of the methodology because these attempts provide a useful source of feedback.

2.2.4.5 Deliverables. The deliverable from this activity would be a series of reports outlining a detailed sequence of steps or procedures for incorporating human engineering into the system development process and which augment and are consistent with existing development methodologies. The reports would contain a discussion of major issues at each step and will suggest tools for facilitating or enforcing implementation of the methodology. These reports would enable the consistent implementation of the methodology, at least on a trial basis.

2.2.5 Detailed Activity 1.5: Design Guidelines

2.2.5.1 Purpose. Design guidelines provide one means of summarizing current knowledge concerning the characteristics of good interface design. As such, they constitute an important first step towards bringing engineering discipline into the interface design process.

Fortunately, a fair amount of work has already been done to compile design guidelines under both industry and DoD sponsorship. All further effort at guideline development should build on and expand existing work rather than start from scratch. There is a definite need to gain experience in actually applying the guidelines to assess their usefulness. Engel and Granda (1975) proposed an early set of guidelines which has served as the basis for later efforts. The Lockheed Missiles and Space Company (1982) has a very good, quite comprehensive set of guidelines. Smith (1982), under contract with the Air Force (Electronic Systems Division), has compiled a total of 580 guidelines broken down into six functional areas (data entry, data display, sequence control, user guidance, data transmission, and data protection).

The effort to develop useful guidelines is a continuing one. Existing guidelines would be expanded to fill in gaps and would be reviewed, refined, modified and, in some deleted. Guidelines, should also be evaluated are made to apply them and as more is known about the principles of interface design.

2.2.5.2 Premises. It is assumed that a number of aspects of the physical (hardware) interface are well understood in terms of their effect on the user. These include such characteristics as viewing angle and distance, keyboard location, and glare. In fact, there are fairly comprehensive military standards for the design of physical equipment (MIL-STD-1472C) which has been the focus of previous

human engineering work. Much less is known about the logical interface, that is, about the principles underlying effective information display and transfer. A basic premise underlying this activity is that further guideline development (and further research) should focus on these conceptual, communication-related issues.

This should be a continuation of current work. There are important questions which must be addressed concerning the best organization and form of guidelines. As noted earlier, Smith (1982) has organized guidelines into major functional areas such as data entry, data display, and sequence control. Each functional area begins with a list of the basic objectives or goals for that area. For example, the objectives for the guidelines related to data entry include such things as "minimized input actions by user" and "low memory load on user." These basic objectives may help the designer in cases which are not covered by specific guidelines since they provide at least some general guidance. As another example, the guidelines used by Lockheed include positive (DO's) and negative (DONT's) examples for each guideline. This is a useful vehicle for explaining each guideline by making more concrete. The issue of how to make the guidelines most useful needs to be addressed.

2.2.5.3 Description. This activity represents a continuation and integration of current efforts. It would survey and build upon these efforts, addressing issues regarding the most useful organization and content of the guidelines. This activity would focus on the logical or conceptual aspects of human-computer interaction. The guidelines would be publicly reviewed. The development, review and refinement of the guidelines would extend over the entire life of the Software Initiative.

2.2.5.4 Coordination. It is important to recognize that guidelines represent only one part of a more general set of activities for incorporating human engineering into system development. As noted

earlier, the general organization of the guidelines must permit a straightforward transition between prior activities and work products (requirements analysis and functional specifications) and later activities (the design of a specific interface). Thus, the organization of the guidelines cannot be determined in isolation from the development of a general human engineering methodology (Detailed Activity 1.4). Thus, there should be a close interaction between the developers of the guidelines and the developers of the general methodology.

There should also be a close interaction between the developers of the guidelines and the contractors involved in the early efforts (Years 0 through 3) to provide human engineering assistance to the Support Systems Task Area (Subtask 2). Attempts to apply existing guidelines should be closely monitored by the developers of the guidelines.

2.2.5.5 Deliverables. The deliverable for this activity would be a usable set of guidelines focusing on the logical characteristics of the user-system interface. The guidelines would be presented in a structure which is immediately usable by interface designers, at least, in conjunction with the general human engineering methodology.

2.2.6 Detailed Activity 1.6: Supporting Tool Development

2.2.6.1 Rationale. The designer of the user interface would require a unique set of tools - that is, a special support environment - to aid in the design, evaluation, and implementation of the user interface.

2.2.6.2 Inputs. This activity is dependent upon the development of the general methodology. The developers of the methodology will suggest sets of tools. These suggestions should be reviewed and prioritized.

2.2.6.3 Description. This activity would involve the development of an integrated, extensible set of tools for facilitating or enforcing the consistent application of the human engineering methodology. Whenever possible, existing prototypes and production tools should be used. While the specific set of tools would depend upon the methodology, possible candidates include the following:

Tools for Requirements Analysis and Functional Specifications:

- operational requirements checklist
- checklist of functional capabilities

Design Tools:

- tools for rapid prototyping
- design consistency checks
- design documentation generator
- usability evaluator
- error message data base and vocabulary control
- grammars
- on-line structured data base containing guidelines

Evaluation Tools:

- tools to measure complexity of the interface
- tools to determine adherence to guidelines
- data collection tools for monitoring user performance (error frequency, command usage, etc.)

Documentation Tools:

- Petri-nets
- transition diagrams

Implementation Tools:

- text formatters
- graphics formatters
- aids for software generation
- tools to verify consistency with design

The tools (or at least working prototypes) would be turned over to the Support Systems Task Area as they become available. The above list represents only a sampling of possible tools. There is a clear need to integrate those selected for development within the broader context provided by the methodology. The sheer number of possible avenues for investment combined with the need for integration and focus serve to underscore the crucial requirement for evaluation and prioritization (Detailed Activity 1.3).

2.2.6.4 Coordination. The tool development activity is dependent on and must follow the development of the general methodology. The tool development must be closely coordinated with the Support Systems Task Area since the tools will be integrated into the support environment.

2.2.6.5 Deliverables. The deliverables would consist of tools for designing, implementing, and evaluating the user interface along with supporting documentation.

2.2.7 Detailed Activity 1.7: Methodology and Tools Applied

2.2.7.1 Rationale. Much can be learned from attempts to apply the human engineering methodology and available tools to a few selected projects prior to recommending their widespread use. This would provide useful feedback by uncovering deficiencies and gaps. In addition, the selected projects would serve as models for later applications of the methodology. This approach to methodology intro-

duction has been successfully applied by Britton and Parnas (1981) who used the development of the A-7E onboard flight software as a model for applying the principle of information hiding and other techniques.

2.2.7.2 Inputs and Premises. The methodology should be available before development of a complete tool set. Additional tools (or working prototypes) would be applied as they become available. It would be important to select several different types of application systems in order to assess the generality of the methodology. Projects should be selected which will be completed within a period of time to allow for further refinement of the methodology and tools (within one year).

2.2.7.3 Description. Three or four development projects would be selected for application of the human engineering methodology and tools (including the design guidelines). The developers of the methodology would be available for consultation throughout the development phase of each project. The application of the methodology would begin at requirements analysis and would extend into the operational phase of each system.

2.2.7.4 Coordination. This activity must be closely coordinated with the Support Systems Task Area. There also must be a close coordination between the activities involved in refining and enhancing the methodology and tools (Detailed Activities 1.5 and 1.8) and these application attempts.

2.2.7.5 Deliverables. For each of the selected development projects, the deliverables would include work products from each major phase of the development cycle (requirements, specification, design, evaluation, implementation, and field operation). These would serve as models for later projects. The specific work product for each phase should be defined by the human engineering methodology.

In addition, a report should be delivered for each project which summarizes the difficulties and successes experienced in applying the methodology and tools and which suggests specific enhancements and refinements.

2.2.8 Detailed Activity 1.8: Methodology and Tools Enhanced

2.2.8.1 Purpose. The purpose of this activity is to continue to refine and enhance the methodology as a result of increased knowledge and experience. These enhancements should be reflected in corresponding extensions to the tool set. In addition, prototype tools would be refined.

2.2.8.2 Description. The methodology and tools would be continually refined and enhanced as a result of feedback from experiences in applying the methodology and tools, and results from further experimentation and research (Subtask 3).

2.2.8.3 Coordination. Attempts to apply the methodology to selected development projects must be closely monitored. In addition, enhancements to the methodology should incorporate results obtained from ongoing research efforts (Subtask 3).

At a higher level, refinements and enhancements to the methodology must be closely coordinated with the Support Systems Task Area. As new tools are developed and become available, they would be included within the tool set maintained by the Support Systems Task Area.

2.2.8.4 Deliverables. The deliverables for this activity consist of periodic revisions and enhancements to the methodology and tool set (e.g. every 9 months).

2.3 Human Engineering of the Support Environment

2.3.1 Purpose

As noted earlier, the Human Engineering Task Area is concerned not only with the end users of embedded systems but with the personnel involved in software development and support activities. It is equally important that the user interface to their computer systems and software tools be well engineered. While the Support Systems Task Area is directly responsible for developing and maintaining the automated support environment, the Human Engineering Task Area should play a major role in the design of the user interface for that environment. The Human Engineering Task Area must also assure that the developments in the Application-Specific Task Area can be and are consistent with each other and with this interface.

There are two activities which present important opportunities for Human Engineering. One of these involves the selection (or design) of a prototype workstation (the hardware interface). The additional capabilities provided by a dedicated processor with wide bandwidth I/O will allow for powerful modes of human-computer interaction including the use of multiple windows, bit-mapped displays, multi-tasking from a terminal, pop-up menus and various pointing devices. Human Engineering should play a role in analyzing the impact of these capabilities on the user. The second activity involves the design of the software interface including command languages, on-line help facilities, and system messages. Since the users of the support environment cover a wide spectrum (designers, programmers, project managers, and clerical support), the actual outcome of these activities is likely to be multiple workstations and multiple user interfaces to the environment.

2.3.2 Inputs

The development of the general human engineering methodology (Subtask 1) would occur in parallel with this activity. Thus, the methodology would not be available in any finished form for the initial design of the user interface. Nevertheless, it is essential to incorporate human engineering principles into these early environment efforts. There are existing guidelines as well as various human engineering concepts which should prove useful on an interim basis.

2.3.3 Description

This subtask involves the design and implementation of the user interface for the support environment and the "generic" Application Specific Environment. This, in turn, includes the design or selection of a prototype workstation (the hardware interface) and the design and implementation of the software interface. For both activities, this would include the formation of an interim methodology and the definition of specific work products. The methodology would include procedures for:

- o analyzing user requirements and relating these to the necessary functional capabilities of the user interface,
- o applying existing design guidelines, and
- o prototyping/simulating the interface combined with experimentation to decide among alternatives.

2.3.4 Coordination

These activities must be closely coordinated with the Support Systems Task Area. In fact, the designers of the user interface would be a part of the support environment development team. It is also important to coordinate these activities with the development of the methodology and tools (including guidelines) since this represents an important source of early experience and feedback.

2.3.5 Deliverables

The deliverables would consist of a prototype workstation and the software interface for the environment plus intermediate work products.

2.3.6 References to Milestone Charts

The major subtask is shown in Figure 1. The detailed activities are shown in Figure 3.

2.4 Description of Detailed Activities for Subtask 2

2.4.1 Detailed Activity 2.1: Prototype Workstation

Prototype Workstation"

2.4.1.1 Purpose. The move toward individual workstations is motivated by several considerations including

- o increased reliability,
- o consistent and rapid response time, and
- o support for powerful modes of human-computer interaction.

There are a number of issues related to workstation design which would impact the user interface and, hence, the performance and satisfaction of the user. Many people believe that the real value of workstation will come from their capabilities to support interactive graphics and other novel forms of human-computer communication (Gutz, Wasserman, and Spier, 1981).

2.4.1.2 Premises and Inputs. The selection or design of an extensible workstation would require input from both the Human Engineering Task Area and the Support Systems Task Area. Human Engineering would focus on the characteristics and capabilities provided to the user by the hardware and their impact on user performance. Support Systems should cover all hardware and software expen-

ditures and should address issues related to the hardware and software resources needed to support the required capabilities of the user interface. Coulouris (1982) provides an excellent analysis of the resources required for a reasonably powerful network of office workstations. His analysis serves as a useful model for the types of decisions which are involved.

Since the workstation would support different groups of users - including project managers and various technical and staff personnel - there may actually be several different configurations of capabilities selected rather than a single workstation.

2.4.1.3 Description. This activity would result in the selection or design of an extensible workstation for each major user group involved in software development and software support activities. For each user group, the required capabilities of the user interface would be identified. In conjunction with the Support Systems Task Area, the hardware and software resources required to support these capabilities would be identified (e.g. bit-mapped color display, minimal processor performance requirements, necessary peripherals, etc.).

Also in conjunction with the Support Systems Task Area and coordination task of the Application Specific Area, a survey would be conducted of existing workstations, both in the marketplace and undergoing development. Prototype workstations would be developed using off-the-shelf components whenever possible. Experimentation with typical users would be carried out whenever useful for deciding among alternative features.

2.4.1.4 Coordination. The final selection or design of the workstation should be the joint responsibility of the Human Engineering and the Support Systems Task Areas and subject to coordination with the Application Specific Task Area. As noted earlier, Human

Engineering should focus on identifying the required capabilities of the user interface and on evaluating the impact of interface characteristics on user performance. Support Systems should provide the hardware and software resources necessary to implement those capabilities. There should also be a close coordination with the Project Management Task Area to insure that the workstation is sufficient for the current and future needs of project managers.

2.4.1.5 Deliverables. The deliverables for this activity consist of the following:

- o an analysis of the major user groups and their activities (Requirements Analysis document),
- o identification of the required capabilities of the user interface (Functional Specifications document), and
- o the results of experiments to evaluate the effects of various interface characteristics on user performance.

2.4.2 Detailed Activity 2.2: Support Environment Interface

2.4.2.1 Purpose. The previous activity would be concerned with the human engineering aspects of the hardware interface between the user and the support environment. The current activity is concerned with the software interface (i.e., all aspects of the dialogue between the user and the environment). There has been a fair amount of discussion about the characteristics which should contribute to a well-engineered software interface for programming support environments (although there has also been a noticeable absence of systematic experimentation). These characteristics include

- o consistency in the user interface across different tools in the environment,
- o the existence of multiple interfaces for different users, and

- o the flexibility to allow users to tailor their own command sequences, abbreviations, menu shortcuts, etc.

2.4.2.2 Premises. The Human Engineering Task Area should be responsible for designing all aspects of the software user interface. This would be carried out in coordination with the Support Systems Application Specific and Project Management Task Areas.

One issue that must be addressed is means for achieving consistency of the interface across different tools in the environment. The Ada Programming Support Environments (APSE) present a particular challenge for this human engineering goal. Given the requirement for a standard KAPSE with movable tools and data, a tool may well be required to have a different user interface (or set of interfaces) for each APSE it is moved to. There are three alternative approaches to achieving the desired portability of tools across environments while achieving consistency of the user interface within an environment. They include:

- o re-engineering the tool for each APSE (an obviously) expensive process over the long term
- o defining and standardizing a minimal user interface (which appears premature at this point in time)
- o defining a technique or tool that allows the user interface to change with minimal tool change.

This issue clearly needs to be addressed by a collaborative effort between the Human Engineering and the Support Systems Task Area.

2.4.2.3 Description. This activity would involve the design of the software interface for the support environment. This includes all system messages, system command languages, on-line help facilities, and off-line documentation. In addition, it includes the user interface for all tools specifically developed for the support environment. The designers should develop and adhere to a specific methodology which would include the following general types of

activities. (The methodology developed under Subtask 1 would not be available during these early years):

- o an analysis of the users, their activities and goals (including project managers, technical and clerical personnel),
- o early specification of design goals,
- o specification of quantitative acceptance tests of user performance (Shneiderman, 1983),
- o an attempt to apply existing design guidelines,
- o the design of all aspects of the interface prior to implementation, and
- o techniques for the early evaluation and rapid modification of the interface.

2.4.2.4 Coordination. This activity should be carried out in conjunction with the Support Systems Task Area. There should also be a close coordination with the Project Management Task Area which would develop management tools for insertion into the environment and with Task 5 of the Application Specific Area.

This activity should be closely monitored by the developers of the general human engineering methodology (Subtask 1) since it would provide an early source of feedback on successes and difficulties.

There should also be close coordination between this activity and the design or selection of the hardware interface (the workstation) since the software interface would be partially dependent on the capabilities of the hardware.

2.4.2.5 Deliverables. The deliverables for this activity consist of the following:

- o an analysis of the major user groups and their activities (Requirements Analysis document),

- o identification of the required capabilities of the software user interface for each user group (Functional Specifications document),
- o quantitative acceptance tests,
- o interface design documentation, and
- o results from early experimentation and evaluation and documentation of subsequent refinements to the interface.

2.5 Subtask 3: Research Program in Human Engineering

2.5.1 Purpose

For the most part, the products which evolve from the previous two subtasks will be based on experienced judgment, consensus and trial and error rather than on a solid foundation of empirical evidence and general principles. This is because no such foundation exists. While interest in human-computer interaction is increasing, the gaps in our knowledge are much more apparent than any body of established findings.

The research program should establish a foundation for further progress in human engineering which is based on models of user behavior, general principles, and empirical evidence. In keeping with the plan's focus on two primary user groups (end users of embedded systems and software development personnel), the research program would be structured around the following two general topics:

- 1) human factors of embedded systems use
- 2) human factors of software development and support.

The research program would support both basic research efforts aimed at a more fundamental understanding of these topics as well as more immediately applied work to validate design guidelines and other aspects of the human engineering methodology. The research program

would also include the development of initial prototypes of tools to support the methodology.

2.5.2 Description

This subtask consists of two distinct activities:

- 1) review and assessment of current and past research efforts in the relevant areas (i.e., human factors of software development and support, human factors issues in the use of embedded systems), and
- 2) initiation and monitoring of research efforts.

2.5.3 Coordination

The priority areas of the research program should reflect the evaluation and priorities established as part of Subtask 1. It is anticipated that the investigations conducted as part of this research program would yield results which have implications for future software development methodologies, team structures, software tools, etc. It would be the responsibility of the Advisory Panel to insure that these results are fed to the Support Systems Task Area and to other relevant areas.

2.5.4 Deliverables

There are two major sets of deliverables under this subtask:

- 1) a review and assessment of past and current work in human engineering, and
- 2) specific research results.

(The specific suggestions by the Advisory Panel for a research program in human engineering along with updates to these suggestions will be conducted as part of Subtask 1.)

2.5.5 References to Milestone Charts

The subtask is shown in Figure 1. The detailed activities for this subtask are shown in Figure 4.

2.6 Detailed Descriptions of Activities for Subtask 3

2.6.1 Detailed Activity 3.1: Research Review

2.6.1.1 Purpose. This review of current and past research efforts in human engineering would provide a starting point for the initiation of further research. It would help to avoid duplication of earlier efforts while pointing to potentially high payoff areas.

2.6.1.2 Inputs and Premises. This survey should extend previous reviews by Atwood and Ramsey (1979), Shneiderman (1980), Sheil (1981), Reisner (1981), and others. The focus of the review should be on concerns relevant to the Software Initiative (i.e., human engineering of support environments and embedded systems) rather than on the entire domain of human engineering.

2.6.1.3 Description. This would involve a review and critical assessment of past work along with suggestions regarding high payoff areas for further research.

2.6.1.4 Coordination. The report resulting from this review should provide input to the evaluation and prioritization of methodologies and tools conducted as part of Subtask 1.

2.6.1.5 Deliverables. The deliverable for this activity would be a report containing a critical assessment and survey of current and past work in the human engineering of embedded systems and software environments, and in the human factors of software development. The report would recommend specific, high priority areas for future emphasis.

2.6.2 Detailed Activity 3.2: Research

2.6.2.1 Rationale. Longer term advances in human engineering must depend on a more fundamental understanding of human problem solving and human-computer interaction. This activity would establish a foundation for the future by initiating a human engineering research program.

2.6.2.2 Inputs. The research program would be planned by an Advisory Panel. A major input should be the survey and assessment conducted as part of this subtask.

2.6.2.3 Description. The research program would be structured around two main areas:

- 1) the human factors of embedded systems use, and
- 2) the human factors of software development and support (including the evaluation of software tools and aids for individuals and for teams).

It would encompass both long-term, basic research efforts aimed at a more fundamental understanding of human problem solving and human-computer interaction as well as more immediately applied work to validate specific principles underlying the human engineering methodology and to evaluate the effect of specific tools. It would also include the initial development of prototype tools and aids (particularly those with a longer-term payoff).

Shneiderman (1983) has suggested a number of areas for research and experimentation. The following list and descriptions are taken verbatim from his paper and represent just a subset of his suggestions. They are included here to point out the richness of this area for experimentation. This very richness underscores the need for the activities in Subtask 1.3 in setting priorities and in insuring that the research activities address the needs of the Task Area.

- 1) Response time, display rates, and operator productivity - many computer professionals believe in the simple principle that faster is always better. There is evidence from several IBM studies and other sources that programmers are more productive when system response time is kept within the one second range or even faster. On the other hand isolated studies have shown that in some business decision making tasks, computer assisted instruction, complex order entry, and introductory sessions with novices, rapid performance leads to poorer learning, less effective decisions, higher error rates, and occasionally decreased satisfaction. A thorough study of multiple tasks with a variety of user communities would shed light on which situations would be improved with shorter response times or faster display rates. Understanding psychological issues of short-term memory load, decision making strategies, and information overload would help in preparing design guidelines for system implementers.
- 2) Menu selection - menu selection is offered on many systems for novice users, but there is little data to support design guidelines. The content, number, placement, and phrasing of menu choices could be studied with attention to titling of menu frames, effectiveness of instructions, availability of type-ahead strategies or menu shortcuts, backtracking, and graphic design to show hierarchical organization. Much progress could be made in this area with modest experimental efforts. There is also an opportunity to investigate software architectures for menu management systems, which dramatically reduce the amount of code while permitting end users to develop and maintain their own menus.
- 3) Command Languages - this traditional style of interaction is another excellent candidate for research to understand the importance of consistency in syntactic format, congruent pairings of commands, hierarchical structure, choice of familiar command names and parameters, suitable abbreviated forms, automatic command completion, and interference from multiple routes to accomplish the same task. The impact of response time and novel hardware display and entry devices on the command set is another worthy topic.
- 4) Graceful evolution - although novices may begin with menu selection, they may wish to evolve to faster or more powerful facilities. Methods for smoothing the transition from novice to intermittent knowledgeable to frequent expert could be studied. The differing needs of novice and experts

in prompting, error messages, online assistance, display complexity, locus of control, pacing, and informative feedback need investigation.

- 5) Specification and implementation of interaction - most interactive systems are constructed with traditional procedural languages but novel techniques could reduce implementation times by an order of magnitude. Specification languages and dialogue management systems have been proposed and some commercial packages are available. Advanced research on tools to aid interactive systems designers and implementers might have substantial payoff in reducing costs and improving quality.
- 6) Direct manipulation - graphical interfaces in which the user operates on a representation of the objects of interest are extremely attractive in computer assisted design and manufacturing, video games, database query, electronic spreadsheets, display editors, etc. Empirical studies would refine our understanding of what is an appropriate analogical representation and the role of rapid, incremental, reversible operations.
- 7) Online assistance - although many systems offer some help or tutorial information online, there is limited understanding of what constitutes effective design for novices, intermittent knowledgeable users, and experts. The role of these aids and online user consultants could be studied to assess their impact on user success and satisfaction. The utility of a separate display or window for assistance or tutorials should be contrasted with the common approach of entering a separate subsystem which displaces the current display of work.
- 8) Program documentation - many organizations have standards for internal and external documentation, but realistic evaluations of effectiveness are rare. Comprehensive trials of documentation style for control flow, data structures, module interfaces, concurrency, and real time constraints would produce guidelines to practitioners and insights to the cognitive processes of program comprehension. A major beneficiary of these results would be program maintenance organizations.

2.6.2.4 Coordination. An Advisory Panel would review the research efforts. The results would be used to further enhance the methodology (Subtask 1).

2.6.2.5 Deliverables. The deliverables would consist of periodic research reports detailing results.

2.7 Subtask 4: Evaluation of Human Engineering Impact

2.7.1 Purpose

The Software Initiative is focused on the development and support of embedded computer systems. A major objective of the Human Engineering Task Area is to improve the user interface for these systems. Since embedded systems development and support are, themselves, human-intensive activities, a second major objective is to improve the human engineering of the software support environment along with the entire software process. The purpose of this subtask is to assess the impact of the Human Engineering Task Area relative to each of these major objectives. This presents a challenge because the Human Engineering Task Area cuts across a number of different applications and user groups. While the general objective is to improve the user interface for embedded systems and for the support environment, this actually translates into different design goals for different applications (and for different user groups within an application). For example the user-interface characteristics which are critical for a weapon system may differ substantially from those which are important for the support environment. Yet, it will be necessary to assess the effectiveness of this task area with respect to these specific goals.

2.7.2 Inputs

A major input to this subtask would be the survey of embedded applications which should be conducted as part of Subtask 1. This

survey would provide a foundation for the measurement and analysis activities of this subtask. This subtask would expand the general framework provided by the survey by formulating specific measurable goals for each application. The automated support environment and, to the extent feasible, the entire process of software development and support should be included in this effort.

2.7.3 Coordination

The general approach taken for this evaluation activity would be to establish human engineering goals for each embedded application and for the different user groups of the support environment. Using these goals as input, measures would be identified to assess progress toward each goal. The effective execution of this subtask would require coordination with both the Measurement Task Area and the Support Systems Task Area. The Measurement Task Area should provide support in the selection of measures for each category. The Support Systems Task Area should provide support in determining the goals for the various classes of users of the automated support environment. In addition, the evaluation of the effects of alterations or additions to the support environment should be evaluated in coordination with the Support Systems Task Area.

There must also be coordination with the Acquisition Task Area since the execution of the activities underlying this subtask would require software developers and support personnel to design and implement mechanisms for data collection, either through instrumentation of their products or through appropriate simulations or user surveys.

2.7.4 Deliverables

The deliverables for this subtask would consist of:

- o a report specifying the human engineering goals for each of the major embedded applications and for the automated support environment,
- o a report identifying measures related to those goals,
- o a report containing an analysis of the feasibility of collecting the necessary measures along with the identification of alternatives (e.g., surveys when objective field data cannot be obtained),
- o a series of reports outlining the requirements for data-collection activities within each category (for use by software development and support personnel),
- o a series of reports containing analyses of the data collected and interpretation of the results in terms of progress toward the stated goals.

2.7.5 Reference to Milestone Chart

This subtask is shown in Figure 1. The detailed activities are shown in Figure 5.

2.8 Description of Detailed Activities for Subtask 4

2.8.1 Detailed Activity 4.1: Human Engineering Goals

2.8.1.1 Purpose. In the development of a given system, it is important to define the human engineering goals for the system at an early point and to carry out a specific set of activities to insure that the development is converging on these goals. The purpose of the human engineering methodology (Subtask 1) is precisely that: to provide the tools and mechanisms for explicitly establishing goals and insuring that they are met. The success of any system can be measured by the extent to which the stated goals are, in fact, met.

On a larger scale, the success of the Human Engineering Task Area would be determined by defining explicit goals and then measuring the extent to which they are met. The purpose of this activity is to establish these goals.

2.8.1.2 Inputs and Premises. This activity would use the survey of DoD embedded systems (from Detailed Activity 1.1) as its major input. The survey would describe the major human factors issues arising in each application. A basic premise underlying this subtask is that there is a high degree of commonality in the required human engineering characteristics within each application and within each major user group of the automated support environment (e.g., project managers vs. technical personnel).

2.8.1.3 Description. This activity would involve the establishment of explicit goals for each major application of embedded systems within DoD and for each major user group of the support environment. To the extent feasible, it would also include the establishment of human engineering goals for the entire process of software development and support. These goals are likely to be expressed in terms of the following dimensions of user performance:

- o ease of initial learning
- o efficiency or speed of steady-state use
- o error rate/accuracy
- o level of user satisfaction.

2.8.1.4 Coordination. This activity should be used as input to Detailed Activity 4.2. It would also provide useful input in the development of the human engineering methodology (Subtask 1) by providing a catalog of goals for general classes of systems.

2.8.1.5 Deliverables. The deliverable for this activity would be a report outlining the human engineering goals for each major DoD embedded application and for each major user group of the automated support environment.

2.8.2 Detailed Activity 4.2: Measures and Feasibility

2.8.2.1 Purpose. The purpose of this activity is to identify the measures which should be used to evaluate the extent to which the human engineering goals are being met for each application. It is not feasible to collect and analyze information on all conceivable aspects of the system characteristics, system usage, and user performance. Rather, the measurement activity must be selective and directed toward the human engineering goals identified in the previous activity.

One necessary part of this activity would be an analysis to determine the feasibility of collecting the measures which are identified. In some cases, the collection of various measures of system usage is straightforward such as on a large host computer where computational power is not at a premium. It should, for example, prove feasible to collect relevant data on the use of the automated support environment. There are certainly many cases, however, in which instrumentation in this fashion is not feasible. For many embedded systems, it will be impractical to sustain the overhead required to collect data. In these cases, alternative techniques must be employed. These may involve simulation of the field environment or surveys assessing the reactions of the end users.

2.8.2.2 Inputs. The major input to this activity would be the catalog of goals generated under the previous activity (4.1).

2.8.2.3 Description. This activity would involve the identification of measures to assess the extent to which the human engineering goals are being met. An important part of this activity would be a feasibility study to identify measures which can be obtained in a cost-effective manner and measures which must be obtained by indirect means. The feasibility analysis would also establish the scope of the data-collection effort for each category (i.e., all units vs. a

representative subset). In many cases, it would be necessary to collect data on a trial basis as part of determining feasibility.

2.8.2.4 Coordination. This activity should be carried out with support from the Measurement Task Area in identifying measures, in analyzing feasibility and in insuring coordination and integration of all measurement activities. This activity would also be carried out with support from the Support Systems Task Area in determining the feasibility of collecting measures on the usage of the automated support environment.

2.8.2.5 Deliverables. The deliverable for this activity would consist of a report identifying measures for each major embedded application and for each major user group of the automated support environment.

A second deliverable would consist of a report containing a feasibility analysis for each measure and identification of alternatives when the collection of objective data from the operational use of a system is not feasible.

2.8.3 Detailed Activity 4.3: Data-Collection Requirements

2.8.3.1 Purpose. In addition to establishing human engineering goals and identifying relevant measures, it would be necessary to explicitly describe the data collection requirements so that developers can carry out the necessary instrumentation. When instrumentation of the end product is not feasible, it may still be feasible to instrument simulators. At the very least, it would be necessary to tailor general surveys to particular systems.

2.8.3.2 Inputs and Premises. This activity would depend on input from the previous activity (4.2) which involves the identifica-

tion of measures for each major application of embedded systems and for each user group of the automated support environment.

This strategy assumes that it would be the responsibility of the developers of a system to put mechanisms for data collection into place. In some cases, this would involve instrumenting the product. In other cases, it might involve instrumenting a simulator or administering surveys in the field. In some cases, it may even involve all three of these activities. The actual data collection would be conducted by the personnel responsible for operational support. The Human Engineering Task Area cannot, within any reasonable financial limit, take on the job of actually instrumenting systems or carrying out the data collection. It could and would give explicit guidance to enable software developers and support personnel to carry out these activities.

2.8.3.3 Description. This task would involve the generation of an explicit set of data collection requirements for each major category of embedded system and for the support environment. These requirements would specify whether actual instrumentation to obtain objective data from the field is required or whether an indirect means of obtaining the relevant data is more cost effective. These requirements would contain sufficient detail to enable developers to implement the relevant data collection mechanisms. Mechanisms for delivery of the data for analysis and interpretation should also be specified.

2.8.3.4 Coordination. This activity must be coordinated with the Acquisition Task Area to insure that the requirements for data collection are written into RFP's. (Whenever possible, this would include not only new systems but existing systems undergoing modification.) This activity must also be coordinated with the Support Systems Task Area which would provide assistance in describing the data collection requirements for automated support environments.

2.8.3.5 Deliverables. The deliverable for this activity would consist of a set of data collection requirements for each major embedded application within DoD and for the support environment. When appropriate, relevant data collection forms (primarily user surveys) would be provided as models to be tailored to specific systems.

2.8.4 Detailed Activity 4.4: Analysis and Interpretation

2.8.4.1 Purpose. This is the core activity to evaluate the impact of the Human Engineering Task Area.

2.8.4.2 Inputs. This activity would depend on completion of the previous three activities within this subtask.

2.8.4.3 Description. This activity would involve the actual analysis and interpretation of the data collected. For each application and for the automated support environment, initial baselines should be established. Over time, changes from this baseline would be tracked. The analysis will focus on changes in user performance, system usage, and user satisfaction as a function of changes in user-interface characteristics. This type of analysis would support model construction which will, in turn, allow for predictions about the effect of proposed changes in the user interface to a system. It should also allow predictions about the usability of a system based on data collected from similar systems. Part of this activity would involve validation of these predictions. This activity would also consist of feeding these results back to the Advisory Panel, to the Support Systems Task Area, to the Measurement Task Area, and to the Acquisition Task Area. This activity would also involve the continual updating of goals and corresponding measures as necessary.

2.8.4.4 Coordination. As results are available, they should be provided to the Advisory Panel. These results would provide significant information about areas requiring greater emphasis and about

suggested cause-effect relationships which should be formally tested as part of the research activities (Subtask 3).

The results should also be fed back to the Support Systems Task Area since they will reflect the effects of changes in the support environment on user performance as well as point to tools which are used infrequently (often a symptom of poor human engineering).

The results should also be fed back to the Measurement Task Area as additional input into the general data base.

Finally, the results should be fed back to the Acquisition Task Area so that as changes occur in the data to be collected, these would be incorporated into the corresponding RFP's.

2.8.4.5 Deliverables. The deliverables for this activity would consist of a series of reports containing the results of specific analyses and assessing the impact of the Human Engineering Task Area on each major embedded application and on the support environment. The reports would point to suggested cause-effect relationships and will suggest areas most in need of improvement.

Reports containing updates to the goals for each application and to the corresponding measures would be delivered as needed.

3.0 ANALYSIS OF BENEFITS

The systematic application of the human engineering methodology developed under this task area should represent an important step in converting the current practices of user-interface design into an engineering discipline. The potential impact of the Human Engineering Task Area is wide ranging. It is intended to increase the usability of support environments for personnel engaged in software development and support. It is also targeted toward improvement of the entire software development process so that the associated procedures and work products are better tailored to human capabilities and limitations. Finally, it is intended to provide software personnel with the procedures and tools to design highly usable systems for their end users. "Increased usability" would be reflected in decreased learning time, greater user efficiency, reduced error rates, and increased user satisfaction.

While quantitative estimates of the potential benefits are difficult to derive due to the lack of baseline data, it is clear that they are substantial. A rough quantitative estimate of these benefits can be derived by the following analysis: The number of people engaged in software development and support for the DoD has been estimated at 100,000 (Everett, 1980). Clearly, even small improvements in the productivity of these personnel can result in huge dollar savings. Assuming that the average cost per manyear is \$75K, each 1% improvement in the efficiency of software personnel could lead to a savings of \$75M. There are at least several times that number of end users of the embedded systems that are built. Assuming a total of 500,000 users of embedded computer systems and, again, assuming that the average cost per manyear is \$75K, each 1% improvement in user efficiency would lead to a savings of \$375M. The actual realization of these savings would depend on the consistent and effective utilization of the human engineering procedures and tools developed.

In the final analysis, the most important benefit may be the avoidance of the loss of human life resulting from user error in the operation of mission-critical systems.

4.0 REFERENCES CITED

Britton, K.M. and Parnas, D.L. A-7E Software Module Guide. NRI Memorandum Report 4702, Washington, D.C., December 1981.

Coulouris, G.F. Designing interactive systems for the office of the future. Behaviour and Information Technology, Vol. 1, No. 1, 1982.

Engel, S.E. and Granda, R.E. Guidelines for man/display interfaces. Technical Report TR 00.2720, IBM: Poughkeepsie, N.Y., December 1975.

Everett, R.R. Yesterday, today, and tomorrow in command, control, and communications. Technology Review, Vol. 85, No. 1, 1982.

Gutz, S., Wasserman, A.I., and Spier, M.J. Personal development systems for the professional programmer. Computer, Vol. 14, No. 4, 1981.

Johnson, D.M. and Martson, H.R. The Role and Tools of a Dialogue Author in Creating Human-Computer Interfaces. Technical Report CSIE-82-7. VPI&SU: Blacksburg, VA, May 1982. (AD 118146)

Human Engineering Design Criteria for Military Systems, Equipment and Facilities. Military Standard MIL-STD-1472C, U.S. Government Printing Office, May 1981.

Human Factors Engineering Criteria for Information Processing Systems. Lockheed Missiles and Space Company, Inc: Sunnyvale, CA, 1982.

Ramsey, H.R. and Atwood, M.E. Human factors in computer systems: A review of the literature. Technical Report SAI-79-111-DEX, Science Applications, Inc: Englewood, CO, September 1979. (NTIS NO. AD A075679)

Ramsey, H.R. and Atwood, M.E. Man-computer interface design guidance: State of the art. In Proceedings of the Human Factors Society

- 24th Annual Meeting, Los Angeles, CA, October 1980.

Reisner, P. Human factors studies of database query languages: A survey and assessment. ACM Computing Surveys, Vol. 13, No. 1, 1981.

Reisner, P. Further developments toward using formal grammar as a design tool. In Proceedings: Human Factors in Computer Systems, Gaithersburg, MD, March 1982.

Sheil, B.A. The psychological study of programming. ACM Computing Surveys, Vol. 13, No. 1, 1981.

Shneiderman, B. Software Psychology: Human Factors in Computer and Information Systems. Little Brown and Co: Boston, MA, 1980.

Shneiderman, B. Human factors of interactive software. Keynote address - IBM Heidelberg Scientific Center Symposium, March 1983. Also to appear in End Users and Human Factors (tentative title), Springer-Verlag Publ.

Smith, S.L. User-System Interface Design for Computer-based Information Systems. Technical Report ESD-TR-82-132, USAF Electronic Systems Division: Hanscom Air Force Base, MA, April 1982.

5.0 BOOKS

(The following list is largely taken from Shneiderman, 1983)

5.1 Journals

Behavior and Information Technology

International Journal of Man-Machine Studies

ACM Computing Surveys (Note Special Issue on "The Psychology of Human-Computer Interaction, Vol. 13, No. 1, 1981)

Communications of the ACM

Ergonomics

Human Factors

IBM Systems Journal (Note Special Issues on "Human Factors", Vol. 20, No. 2, 1981 and "Man-Machine Interaction", Vol. 20, No. 3, 1981)

IEEE Computer

IEEE Transactions on Systems, Man, and Cybernetics

Journal of Applied Psychology

5.2 Books

Badre, A. and Shneiderman, B. (Eds.), Directions in Human-Computer Interaction. Ablex Publishing Co: Norwood, N.J., 1980.

Bailey, R.W., Human Performance Engineering: A Guide for System Designers. Prentice-Hall, Inc: Englewood Cliffs, N.J., 1982.

Cakir, A., Hart, D.J. and Stewart, T.F.M., Visual Display Terminals: A Manual Covering Ergonomics, Workplace Design, Health and Safety, Task Organization. John Wiley and Sons: New York, 1980.

Committee on Human Factors, Research Needs for Human Factors. Commission on Behavioral Social Sciences and Education, National Research Council. National Academy Press: Washington, D.C., 1983.

Curtis, B. (Ed.), Tutorial: Human Factors in Software Development. IEEE Computer Society Press, New York, 1981.

Foley, J.D. and Van Dam, A. Fundamentals of Interactive Computer Graphics. Addison-Wesley Publishing Co: Reading, Mass., 1982.

Guedj, R. (Ed.), Methodology of Interaction, North-Holland Publishing Co: Amsterdam, 1980.

Hiltz, S. R. and Turoff, M. The Network Nation: Human Communication via Computer. Addison-Wesley Publishing Co: Reading, Mass., 1978.

Larson, J.A. (Ed.), Tutorial: End User Facilities in the 1980's. IEEE Computer Society Press: New York, 1982.

Martin, J. Design of Man-Computer Dialogues, Prentice-Hall, Inc: Englewood Cliffs, N.J., 1973.

Mehlmann, M. When People Use Computers: An Approach to Developing an Interface. Prentice-Hall, Inc: Englewood, N.J., 1981.

Nichols, J. (Chair.), Proceedings: Human Factors in Computer Systems, ACM: Gaithersburg, Md., 1982.

Shneiderman, B. Software Psychology: Human Factors in Computer and Information Systems, Little Brown and Co: Boston, Mass., 1980.

Smith, H.T., and Green, T.R.G. (Eds.), Human Interaction with Computers. Academic Press: New York, 1980.

Weinberg, G.M. The Psychology of Computer Programming. Van Nostrand

Reinhold, New York, 1971.

5.3 Workshops

Human Factors in Computer Systems -- Current Trends and Recommended Directions. (Sponsored by the Computer Systems Group and Rocky Mountain Chapter of the Human Factors Society.) April 11-14, 1983, Vail, Colorado. Workshop report will be publicly available through the Human Factors Society.

6.0 LINKAGES TO OTHER TASK AREAS

6.1 Linkage to Human Resources

SKILLS - The human engineering of computer-based systems has only recently emerged as an identifiable area of study and application. Practitioners have emerged from a number of different areas including cognitive psychology, computer science, human factors, traditional human engineering, industrial engineering, and ergonomics. There is a growing awareness of the need to establish a general job category that might be called "Human Factors Software Engineer". A person within this general category will concentrate on the interface between a human user and the software and hardware of computer-based systems. It is expected that there will be specialties within this general category such as graphics designers, dialogue authors, and so on. An early task will entail the identification of skill requirements for this Human Factors Software Engineer. This appears to be a task for the Human Resources Task Area with support from Human Engineering.

Two different user communities will benefit from the application of human engineering skills. One community consists of the end users of DoD embedded systems such as pilots, weapons officers, and communications personnel who operate the embedded systems. The other community consists of the personnel involved in software development and support who either produce the software for the embedded systems or produce the automated environments that host the software-related activities and products.

TRAINING - As noted above, the human engineering of computer systems is an area which has emerged from a number of different disciplines. The relevant literature is not easy to locate, coming from many diverse areas. There is currently no complete training curriculum.

In order to establish a body of expertise which matches the skill needs to be defined, a curriculum must be established within the most appropriate discipline (e.g. computer science, human factors). This appears to be a task for the Human Resources Task Area.

CAREER PATHS - Given the identification of the requisite skills and establishment of appropriate curricula, definitions are required of the various jobs to be performed by human factor software engineers. Description of these positions appears to be a task for the Human Resources Task Area.

6.2 Linkage to Project Management

Project managers are responsible for overseeing software development, installation, and support. They prepare a work breakdown for the project, produce project estimates, identify requisite skills, assemble a team, assign, schedule, and monitor work accomplishment. Every project manager wants to build a high quality system that is admired by colleagues, celebrated by users, circulated widely, frequently imitated, within project cost and schedule. Human engineers can contribute to these system design goals when assigned early in the development phase (i.e., during requirements analysis) and retained throughout the project life cycle. When modeling project management, there should be specific coordination with the Human Engineering Task Area in order to identify and insert human factors decision points into the project management process model. In particular, it is essential that project managers know where human engineering concerns rank in the list of project priorities for any given project. These priorities must be explicit to allow for the appropriate allocation of resources.

A second linkage between Project Management Task Area and the Human Engineering Task Area concerns the human engineering of the management tools and methodologies themselves. The Human Engineering

Task Area can assure that the project management tools are also well engineered from the standpoint of the people who will use the tools. This activity is included within Subtask 2 of the Human Engineering Task Plan (selection of workstations and design

6.3 Linkage to Support Systems

There are a number of essential linkages between the Human Engineering and the Support Systems Task Area. In fact, Human Engineering is more closely linked with this area than with any other. Each of the four subtasks underlying the Human Engineering Task Plan depends on coordination with Support Systems. These are described below.

METHODOLOGY DEVELOPMENT (Subtask 1) - It is essential that the activities underlying the development and application of the human engineering methodology be closely coordinated with the Support Systems Task Area. To be successful, the human engineering methodology must result in procedures and work products that are consistent with other system development activities and products. The human engineering methodology must be integrated into the more general development methodology.

HUMAN ENGINEERING OF THE SUPPORT ENVIRONMENT (Subtask 2) - One of the early products of technology consolidation is the planned workstation. The Human Engineering Task Area will work with Support Systems in designing or selecting this workstation. The Human Engineering Task Area will focus on identifying the required capabilities of the hardware interface and on evaluating the impact of these capabilities on user performance. The Support Systems Task Area will provide the hardware and software resources necessary to implement those capabilities.

In addition to participation in the workstation selection, the Human Engineering Task Area will be responsible for designing the

user interface to the software functions of the support environment (e.g., command languages, help facilities, etc.).

RESEARCH PROGRAM IN HUMAN ENGINEERING (Subtask 3) - It is anticipated that the investigations conducted as part of the research program in human engineering will yield results which have implications for future software development and support methodologies, team structures, software tools, etc.

To be useful, these results must be transferred to the Support Systems Task Area.

EVALUATION OF HUMAN ENGINEERING IMPACT (Subtask 4) - The Support Systems Task Area will support Human Engineering in establishing the goals for the classes of users of the automated environment. In the evaluation of the effects of changes in the environment (e.g., addition of tools, changes in the user interface), will be conducted in conjunction with the Support Systems Task Area.

6.4 Linkage to Application Specific

The Application-Specific Task Area will support the development of reusable software components for selected applications. There are a number of human engineering issues related to mechanisms or guidelines for specifying and cataloguing software components. This expertise can (and should) be provided by the Human Engineering Task Area although no such linkage currently exists within the Human Engineering Task Plan.

Different applications will require different human engineering goals in system design. There are two sequential activities within the Human Engineering Task Area which are directed at defining these goals for each major application. Specifically, Detailed Activity 1.1 consists of a survey of DoD embedded systems to identify the major human factors issues. Detailed Activity 4.1 consists of the explicit specification of human engineering goals for each of these

applications. The Application-Specific Task Area can provide useful support in reviewing these activities.

6.5 Linkage to Technology Insertion

It is expected that several human factors software engineers will be resident at the Software Engineering Institute. These individuals, each in a one- to two-year tenure, will conduct studies to evaluate human engineering aspects of the support environment. They will also focus on issues related to the integration of human engineering procedures and tools into the support environment.

6.6 Linkage to Measurement

The Measurement Task Area will insure coordination and integration of all measurement activities within all task areas of the Software Initiative. Specifically, it will support the development, use and refinement of measures to aid in all phases of software development and support. It will also provide support in assessing the effectiveness of the different task areas. The Human Engineering Task Area will coordinate closely with the Measurement Task Area to:

- o identify measures to assess the impact of the Human Engineering Task Area (Subtask 4)
- o identify points in the system development cycle for data collection and other forms of quantitative assessment (Subtask 1)
- o use the results from the above two activities to evaluate and revise human engineering methodologies and tools.

6.7 Linkage to Acquisition

The plan for the Acquisition Task Area includes activities to provide appropriate contractual incentives and guidelines and to identify new technologies enhancing the acquisition process. The tools and methodologies produced by the Human Engineering Task Area will be consistent with these acquisition enhancement goals. Close

coordination is required between the panel to be established in the Acquisition Task Area and the Advisory Panel to be established in the Human Engineering Task Area so that information may be shared as to tools and methodologies of potentially high payback as well as assistance provided for structuring contractual guidelines and incentives.

The evaluation of the impact of the Human Engineering Task Area (Subtask 4) will require close coordination with the Acquisition Task Area since system contractors will be required to develop mechanisms for collecting data about system usage and to carry out this data collection.

6.8 Linkage to Systems

There are no direct linkages between these two task areas.

APPENDIX I

HISTORY OF THE HUMAN ENGINEERING PLAN

The general history of the Software Initiative is outlined in a document entitled "Strategy for a DoD Software Initiative" which appeared in October 1982. The strategy document identifies eight different task areas, one of which is Human Engineering. The primary activities described for Human Engineering in that document include the selection or design of a prototype workstation and the initiation of a research and development program.

A more detailed strategy for Human Engineering was prepared in late November 1982. This detailed strategy built upon the Human Factors section of the Appendix to the earlier strategy document. Additional input was obtained from the responses to the "Candidate R&D Thrusts for the Software Technology Initiative". This more detailed strategy expanded the scope of the task area to include the software interface between the user and the automated support environment (in addition to the workstation activity). It also focused on the need for a general human engineering methodology to incorporate human factors principles throughout the development of an interactive system (whether this system is an automated support environment or an application system). In mid-December, this more detailed strategy was reviewed by a task force of DoD personnel from the component services. Following this review, a subtask to assess the impact of the Human Engineering activities was incorporated into the strategy.

The next major point of review occurred at the Software Initiative Workshop held in February in Raleigh, North Carolina. A panel of leading researchers and practitioners in human factors was assembled for this workshop. The panel consisted of the following members:

Chair:	Carol Morgan	-	NAVSEA (also a member of the Software Initiative Task Force)
Vice-Chair:	Elizabeth Kruesi	-	General Electric (author of the detailed plan)
	Louis Blazy	-	Army Research Institute
	Thomas Landauer	-	Bell Laboratories
	Timothy Lindquist	-	Virginia Tech
	David Lenorovitz	-	Martin Marietta Denver Aerospace
	Donald Monk	-	Air Force Aerospace Medical Research Laboratory
	John O'Hare	-	Office of Naval Research
	Phyllis Reisner	-	IBM
	Sidney Smith	-	MITRE

The purpose of the workshop was to receive comment from a wide segment of the technical community. The Human Engineering strategy was presented at a four-hour session to an audience of approximately 70 people. During that session, a total of nineteen written comments were submitted along with a number of spoken comments and questions. Following the Raleigh workshop, the strategy underwent one major revision incorporating a document entitled "Human Factors Engineering and Software" by Donald Monk of the Air Force Aerospace Medical Research Laboratory. The following points cover the major issues raised by the audience and their impact on the plan.

- 1) Comment: Several people pointed out that the strategy was entirely concerned with the human-computer interface instead of with the human engineering of the entire process of software development and support. Impact: This widening of the scope of human engineering has been incorporated into the latest revision of the strategy.
- 2) Comment: One person suggested that the user's conceptual model of the system be considered. Impact: This is viewed as a worthy research topic with a longer-term payoff. Presumably, it will be considered by the Advisory Panel in planning the research program.
- 3) Comment: One person pointed to the problem of the long timelag in applying research results. In general, this problem cuts across the entire Software Initiative. Impact: This timelag may be lessened by initiating a focused

research program that is directed at solving the specific set of problems falling within the realm of the Initiative. For this reason, the research program for Human Engineering will consist of a focused set of activities.

- 4) Comment: Several people pointed out that most embedded systems do not have a CRT interface yet the strategy seemed to be directed towards a terminal interface. It was suggested that other types of I/O be explicitly mentioned such as voice, tactile, and analog displays. Impact: It was agreed that this was under-emphasized in the original strategy. The focus of the strategy has now shifted to include the end-user of embedded systems.
- 5) Comment: One person commented on the importance of measurement to the goals of the Human Engineering Task Area. Mechanisms are needed for obtaining feedback from the field use of end products. Impact: The support environment will be instrumented as part of the Measurement Task Area. The problems involved in obtaining feedback about the use of embedded systems will be addressed by Subtask 4 of the current strategy.
- 6) Comment: Several people pointed to the severity of the consequences of poor human engineering of tactical embedded systems. Impact: This has been mentioned in the current revision.
- 7) Comment: One person commented on the need for validating the human engineering methodology. Impact: This was interpreted in two ways, both of which are important and necessary. The methodology must be applied and data collected to show that the system is actually better as a result of having been developed under the methodology. Subtask 4 of the current plan is directly concerned with the need to collect such data. Secondly, mechanisms must be developed, either through acquisition, management, or through the use of tools, to insure adherence to the methodology. The necessary linkages must be set up with the Management, Acquisition, and Support Systems Task Area to insure such adherence.
- 8) Comment: One person commented on the lack of a clear responsibility for the measurement aspect of Human Engineering. It was unclear whether it should lie with the Measurement Task Area or with Human Engineering. Impact: This was clarified in the current strategy by assigning that respon-

sibility to Human Engineering with support from the Measurement Task Area.

- 9) Comment: The point was made that prototypes can be very useful in determining system requirements. Impact: It is assumed that the use of prototypes belongs under Subtask 1 of this plan and will be addressed by the Advisory Panel. It also seems to overlap with the Support Systems Task Area.
- 10) Comment: One person suggested that Human Engineering should export its expertise in evaluation and experimentation into the other task areas. Impact: It is recognized that many of the people involved in human factors activities are trained in experimental design and statistical analysis. This is, however, the responsibility of the Measurement Task Area although a synergistic link between the two areas is certainly expected.
- 11) Comment: The panel felt that there is a clear need for a steering group to be responsible for the focus of the methodological activities. This includes assessing the currently available techniques and tools and guiding the selection of further activities. Impact: In the current revision, these functions have been incorporated into the previously planned Research Advisory Panel. The establishment of this panel is now a part of Subtask 1 (Methodology Development).
- 12) Comment: There were several issues concerning the human engineering of the support environment. The panel noted that there is essentially no work on the human engineering of automated environments for software development. There is much talk about human engineering which focuses on discussions of the use of graphics, various pointing devices and so on. However, no one appears to address basic principles of interface design or conduct systematic experimentation, both of which fall within the domain of a true human engineering discipline. In addition, it was pointed out that automated support environments present special problems for human engineering. Impact: The plan has been revised to include a discussion of the need to address the issue of maintaining a consistent user interface across tools while allowing for portability of tools across environments.
- 13) Comment: One panel member expressed concern about the implementation of the Advisory Panel. The following is a direct quote from this member. "This panel has to be thought through more carefully. If it is contracted-out to

an external-DoD organization, its members and their organizations cannot participate in the contracted work. It is likely that the experts needed to support the work of the panel will come from different organizations so that it would be difficult to sole-source one organization to this work. The connection between the panel and subsequent contracts to do the work needs to be established and the role of DoD clarified. It does not seem proper to defer such expenditures to an external organization; some DoD management responsibilities would seem obligatory. I would suggest a DoD-managed panel (on a part-time, as needed basis) with the actual work delegated to paid consultants (experts). Some mechanism for their periodic reporting and interaction has to be established." Impact: Since this is a technical plan and not an implementation plan, that issue has not been addressed. It is clearly an important issue which will have to be addressed along with other implementation and legal issues.

END

FILMED

6-83

DTIC