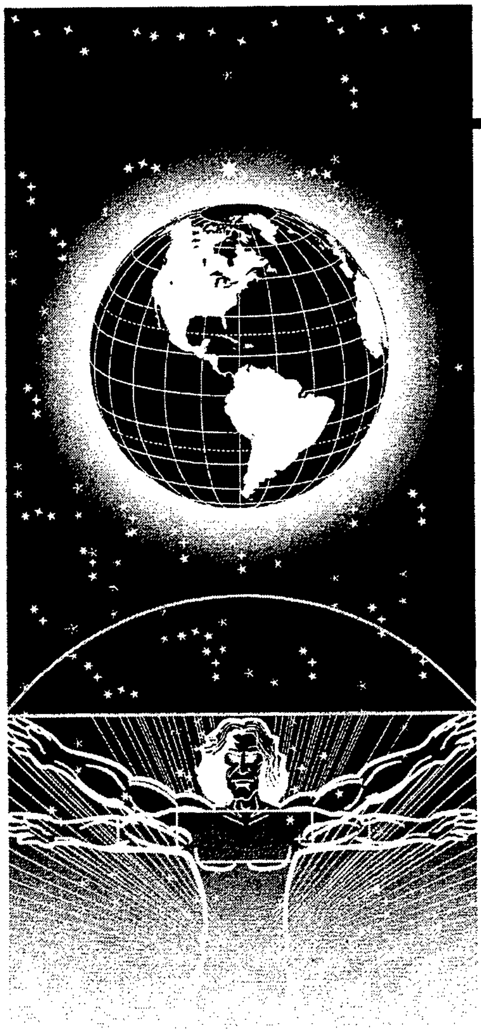
AFRL-HE-WP-TR-2000-0009



UNITED STATES AIR FORCE RESEARCH LABORATORY

THE COMBAT AUTOMATION REQUIREMENTS TESTBED (CART) TASK 1 FINAL REPORT: IMPLEMENTATION CONCEPTS AND AN EXAMPLE

Bryan E. Brett
Jeffrey A. Doyal
David A. Malek

SCIENCE APPLICATIONS INTERNATIONAL CORPORATION
4031 COLONEL GLENN HIGHWAY
BEAVERCREEK OH 45431

Edward A. Martin
David G. Hoagland

HUMAN EFFECTIVENESS DIRECTORATE
CREW SYSTEM INTERFACE DIVISION
WRIGHT-PATTERSON AFB OH 45433-7022

JANUARY 2000

INTERIM REPORT FOR THE PERIOD OCTOBER 1998 TO SEPTEMBER 1999

Approved for public release; distribution is unlimited

Human Effectiveness Directorate
Crew System Interface Division
2255 H Street
Wright-Patterson AFB, OH 45433-7022

NOTICES

Please do not request copies of this report from the Air Force Research Laboratory. Additional copies may be purchased from:

National Technical Information Service
5285 Port Royal Road
Springfield, Virginia 22161

Federal Government agencies and their contractors registered with the Defense Technical Information Center should direct requests for copies of this report to:

Defense Technical Information Center
8725 John J. Kingman Road, Suite 0944
Ft. Belvoir, Virginia 22060-6218

TECHNICAL REVIEW AND APPROVAL

AFRL-HE-WP-TR-2000-0009

This report has been reviewed by the Office of Public Affairs (PA) and is releasable to the National Technical Information Service (NTIS). At NTIS, it will be available to the general public.

This technical report has been reviewed and is approved for publication.

FOR THE COMMANDER

MARIS M. VIKMANIS
Chief, Crew System Interface Division
Air Force Research Laboratory

REPORT DOCUMENTATION PAGE	*Form Approved* *OMB No. 0704-0188*

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE January 2000	3. REPORT TYPE AND DATES COVERED Interim, October 1998 - September 1999

4. TITLE AND SUBTITLE

The Combat Automation Requirements Testbed (Cart) Task 1 Final Report: Implementation Concepts and an Example

5. FUNDING NUMBERS

C: F41624-98-C-6012
PE: 63231F
PR: 2830
TA: 29
WU: 10

6. AUTHOR(S)

* Bryan E. Brett, * Jeffrey A. Doyal, * David A. Malek
** Edward A. Martin, ** David G. Hoagland

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

* Science Applications International Corporation (SAIC)
4031 Colonel Glenn Highway
Beavercreek OH 45431

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

** Air Force Research Laboratory, Human Effectiveness Directorate
Crew System Interface Division
Air Force Materiel Command
Wright-Patterson AFB OH 45433-7022

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

AFRL-HE-WP-TR-2000-0009

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release; distribution is unlimited

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

Military weapon systems are built to satisfy requirements levied by the warfighter. These weapon systems are all manned in some sense, yet tools for quantifying the effectiveness with which a crewstation must support operator performance are lacking. The Combat Automation Requirements Testbed (CART) program will provide human performance modeling methods and tools for generating objective, performance-based crew system requirements. CART will integrate readily with techniques currently used within the acquisition process to generate system requirements. CART will enable acquisition analysts to consider the crew system at the same time other key system components are being evaluated during the system definition activities that occur early in the acquisition process. This report begins with a discussion of why performance-based crew system requirements are needed, followed by a description of how modeling and simulation are currently used in the acquisition process to generate system requirements. Next, it describes the modeling requirements for developing performance-based crew system requirements and CART's human performance modeling architecture. The discussion then turns to a notional example of how CART would be applied to generate a set of crew system requirements. Finally, the last section discusses the costs associated with applying CART's capabilities and the benefits to be derived therefrom.

14. SUBJECT TERMS

Human Performance Model, Constructive Simulation Analysis of Alternatives, Use-Centered Design, Human-Centered Design, Human-System Interface, Human Behavior Representation

15. NUMBER OF PAGES

73

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT UNCLASSIFIED	18. SECURITY CLASSIFICATION OF THIS PAGE UNCLASSIFIED	19. SECURITY CLASSIFICATION OF ABSTRACT UNCLASSIFIED	20. LIMITATION OF ABSTRACT UNL

Standard Form 298 (Rev. 2-89) (EG)
Prescribed by ANSI Std. Z39-18

This page left blank intentionally.

PREFACE

This effort, "Task 1, Crew Systems Requirements Establishment," was conducted under contract number F41624-98-C-6012 with the Crew Systems Development Branch, Crew System Interface Division, Human Effectiveness Directorate of the Air Force Research Laboratory (AFRL/HECI), Wright-Patterson Air Force Base, Ohio 45433-7022, for the period October 1998 to September 1999. Science Applications International Corporation (SAIC), 4031 Col Glenn Highway, Beavercreek, Ohio 45431-7753 was the contractor. Mr. David Hoagland (AFRL/HECI) was the Program Manager. This effort supported Work Unit 28302910, "Combat Automation Requirements Testbed (CART)."

This page left blank intentionally.

TABLE OF CONTENTS

LIST OF FIGURES

1.0 INTRODUCTION

The Combat Automation Requirements Testbed (CART) program provides human performance modeling methods and tools for generating objective, performance-based crew system requirements. This constructive simulation capability will integrate readily with techniques currently used within the acquisition process to generate system requirements. CART will enable acquisition analysts to consider the crew system at the same time other key system components are being evaluated during the system definition and refinement activities that occur early in the acquisition process. Ultimately, CART will provide for more effective crew system designs that reliably achieve mission performance goals, take less time to develop, and have fewer flaws that require subsequent redesign cost and effort.

This report describes the rationale for why CART is needed and how it will be implemented. It begins with a discussion of why performance-based crew system requirements are needed, followed by a description of how modeling and simulation are currently used in the acquisition process to generate system requirements. Next, it describes the modeling requirements for developing performance-based crew system requirements and CART's human performance modeling architecture. The discussion then turns to a notional example of how CART would be applied to generate a set of crew system requirements. Finally, the last section discusses the costs associated with applying CART's capabilities and the benefits to be derived therefrom.

2.0 THE NEED FOR OBJECTIVE, PERFORMANCE-BASED CREW SYSTEM REQUIREMENTS

Despite significant levels of automation applied in modern weapons and command and control systems, human operators still play significant roles in those systems. Consequently, the performance of an operator can have significant impacts on the performance of the entire system and, indeed, on the outcome of a mission. Among the factors that influence operator performance are the tasks assigned to the operator and the design of the crew system that enables performance of those tasks.

Ideally, the crew system design process should produce a crew station that results in a level of operator performance needed to successfully perform a mission. In practice, this is often not the case, and crew systems are fielded with problems that must be corrected. Doyal, Irvin, and Ramer (1995) reported on an evaluation of cursor system gain functions implemented in the original design of the B-2 bomber. The impetus for the study was a problem related to the aircrew's ability to control the cursor used for navigation and target updates. While it represents a small component of the overall crew system, the cursor is a very important element in the employment of B-2 precision guided weapons. As originally implemented, aircrews were unable to accurately and quickly control cursor placement on desired targets and aim points in a radar image. This resulted in errors and delays in target designation that could have an impact on degree of target destruction.

Not all crew system design problems occur as a function of developing completely new systems. Often, capabilities are added to an existing system to enhance or extend mission performance. These enhancements can have operator performance implications. In such instances, the crew system must be modified to fully achieve the mission performance gains offered by the enhancement. Appendix A presents an unpublished study conducted by two of the authors. The study documents how crew system design is typically conducted within the acquisition process (especially, system modifications) and how design errors can occur. The study evaluated a weapon assignment interface that was developed for an attack aircraft to support employment of a new weapon for the aircraft. An important requirement for the interface was the ability to reassign weapons when target sets were changed during the course of a mission. Unique characteristics of the weapon as it was installed on the aircraft made reassignment of weapon a cognitively complex task. The interface originally provided to support the reassignment task generated error rates and excessive performance times that resulted in poor mission performance (up to 33% of weapons were not released against targets in a simulated mission). Once again, a design flaw in a small component of the overall crew system had a major impact on system performance. Correcting the problem required redesign of the interface and subsequent development and implementation of the new design. This is just one example of how relatively minor design flaws can have significant cost as well as performance consequences. Indeed, data from acquisition studies indicate that correcting problems in a fielded system can cost one hundred times more than correcting problems caught during the system definition and requirements phase (Frost and Thomen, 1998).

Beyond the performance of individual tasks, consideration of the complete set of activities required of an operator is important. The operator in complex, modern, military systems must often respond to multiple, simultaneous, competing demands. Performance requirements for a single task cannot be considered in isolation. The task must be understood in the context of other tasks that can be expected to occur in the same time frame. This is particularly important for understanding the time constraints on a task given time required by other tasks. In a broader sense, the issue is one of task allocation. A critical element of crew system design is specification of the operator's role in that system. The objective is to define a set of activities that the operator can reliably and effectively perform. The challenge of effective task allocation and the frequent failures that occur is demonstrated by the tremendous attention given to workload and situation awareness issues in the human factors literature.

The fundamental assumption of CART is that failures in crew system design are rooted, to a large extent, in the crew system requirements generation process. Crew system design occurs within the broader system acquisition process. A systems engineering approach is followed in which design is preceded by requirements. Within the systems engineering process, great emphasis is placed on requirements generation because the requirements define the capabilities and attributes of the system that will make it acceptable to and effective for the user and other stakeholders. Inadequate requirements lead to inadequate design which, in turn, leads to development and implementation of an inadequate system.

Traditional crew system requirements tend to dictate general capabilities and components of the crew station (e.g., some number of displays of a certain size and resolution, certain control devices, information content of displays), but they stop short of specifying how well an operator needs to be able to perform with them. Performance, however, is a critical system attribute for military system users and stakeholders as they have expectations for the level of mission performance a system should be able to achieve. To ensure that mission performance expectations are met, crew system designers need clear, objective performance-based guidance (versus design guidance) on the levels of performance their crew system must be able to support. In the absence of objective performance requirements, crew system design becomes somewhat hit or miss. Also, there are no objective performance criteria against which designs can be tested. Sub-optimal designs can be produced and the flaws can go undetected, often until the system becomes operational -- as was the case with the B-2 cursor example discussed above.

Imagine, however, that the designer of the B-2 cursor control interface was given the following as a requirement: "The B-2 cursor control interface shall enable the operator to designate a target in no more than 5 seconds with an average error no greater than 1.5 pixels from the intended designation point." Design could then be conducted using the best practices and design tools available with reference to clear performance criteria, which could also be used to test the designs that result. Given such a requirement, it is much less likely that a flawed design would be produced and that costly remediation would be required.

Objective performance requirements are of value only to the extent that they have a known association with system and mission performance. Consequently, an essential capability for generating objective requirements is a means for predicting mission performance based on given levels of operator performance. Within CART, constructive human performance models will be used to represent operator performance and predict its impact on mission performance. These constructive operator models will be applied in much the same way that constructive system models are used at present to generate system requirements. Because CART leverages and extends current methods for generating system requirements, the process of generating system requirements will be described before discussing how CART will apply human performance models.

3.0 CURRENT STATE-OF-THE-ART FOR GENERATING SYSTEM REQUIREMENTS IN THE ACQUISITION PROCESS

3.1 Overview of the Acquisition Process

The overall acquisition management process, outlined in Figure 1, consists of five distinct phases separated by milestone decision events that determine whether to proceed to the next phase (Air Force Instruction 10-601, 1998). The initial set of activities, referred to as Pre-Milestone 0 activities, focus on determining whether there is a need for a new system acquisition. Given that a need for a new system is determined, Phase 0 (Concept Exploration) is initiated. In Concept Exploration, alternative concepts for meeting the mission need are evaluated in a formal study process called the Analysis of Alternatives (AoA). Objectives of the AoA process include identifying the advantages and disadvantages of acquiring a new system over modifying an existing system; defining the characteristics needed in the new system; and selecting the preferred alternative(s) to carry into Phase I of the program, Program Definition and Risk Reduction (PDRR). The desirable characteristics of the proposed system, including performance, operation, and support requirements, are then reflected in an Operational Requirements Document (ORD) and a Requirements Correlation Matrix (RCM). During PDRR assessments of the advantages and disadvantages of the preferred system concept(s) continue to be refined with emphasis on life-cycle cost, performance, supportability, and schedule impacts. Upon completion of PDRR, a single most-promising system approach will have been identified. At this point, the program will enter the Engineering and Manufacturing Development (EMD) phase, in which the preferred approach will be translated into a stable, producible, supportable, and cost-effective design. Throughout PDRR and EMD, requirements generation activities change focus. Earlier in the acquisition process, requirements are determined by the government and stated in high-level, operational terms. As the EMD phase is entered, the contractor begins to translate and evolve the high level *operational* requirements stated in the ORD down to specific detailed *design* requirements for the system components. Finally, once the detailed design has been firmed up in EMD, the Production, Fielding / Deployment, and Operational Support phase is initiated. The goal of this phase is to develop an operational capability that meets the identified mission needs.

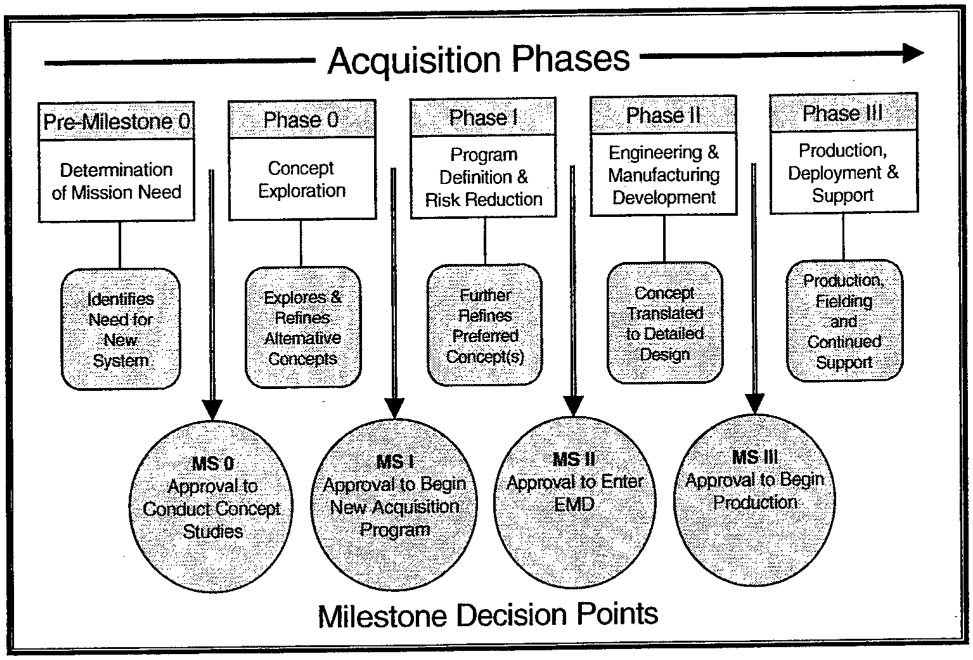


Figure 1. Phases and Milestones in the Acquisition Cycle

3.2 Trade Studies

An integral component of the acquisition process described above, particularly in the early phases, are trade-off analyses (or trade studies). These studies are performed in an effort to continually narrow down the solution space of alternative system concepts under consideration. Figure 2 illustrates the process. Trade studies examine multiple attributes of the proposed systems including predicted mission performance, cost, supportability, and maintainability in an effort to identify a single approach that best meets the cost and performance goals of the acquisition program. Objectives of trade studies include user requirements definition and refinement, attribute tradeoff analyses, and assessments of military worth.

Modeling and simulation is used extensively to support trade studies. During a tradeoff analysis, system concepts are examined in simulation environments to predict the effectiveness, costs and performance associated with each system. Data from these simulation activities are then used to develop curves that help characterize the relative performance impacts associated with a given change in cost. Often, analysts seek to identify bends or knees in the cost-benefit curve. These knees indicate a point of diminishing returns, at which continued performance gains become increasingly expensive. Assuming they fall within the determined cost and performance limits, these points in the curve indicate the most cost-effective solutions, and are translated into system requirements.

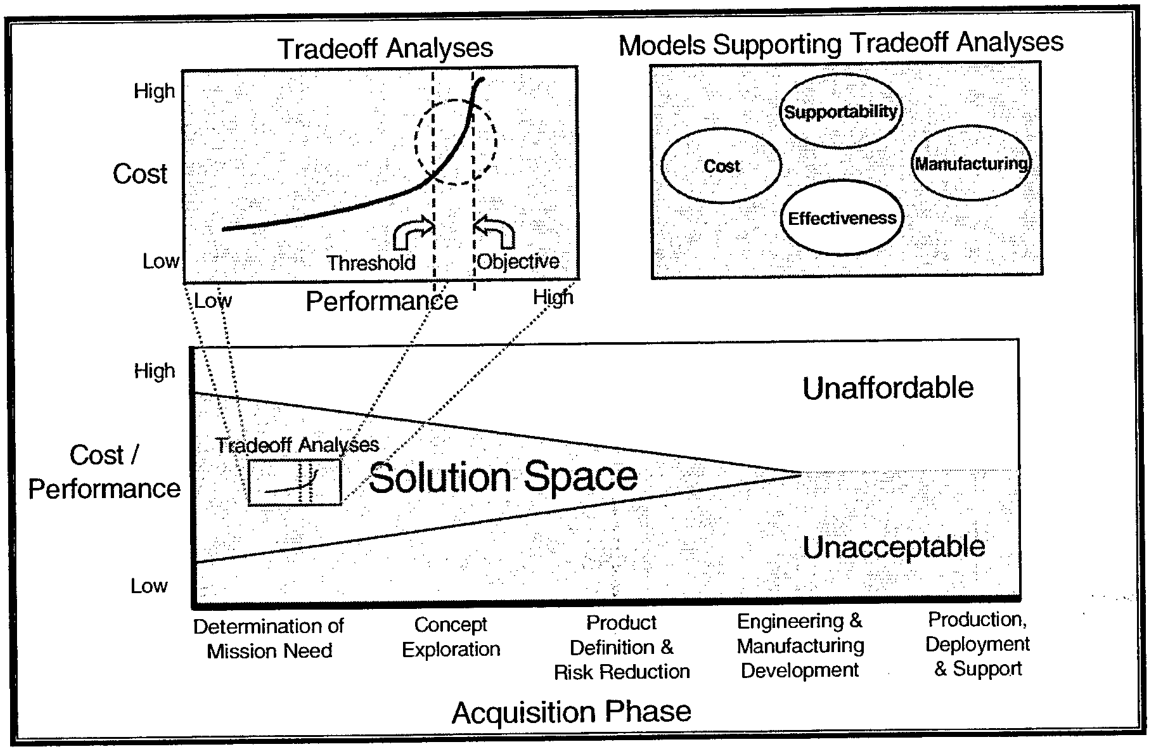


Figure 2. Tradeoff Analyses Throughout the Acquisition Lifecycle

In support of these tradeoff analyses, various types of digital models of systems and subsystems are used -- including cost, supportability, manufacturing, and effectiveness models. Results provided by effectiveness models are used to generate system performance requirements. Depending on the focus of the analysis, effectiveness modeling can involve all levels of the modeling and simulation pyramid presented in Figure 3 (*Air Force Modeling and Simulation: A New Vector*, 1995).

Generation of system requirements, however, generally employs mission and engagement level simulations. CART human performance models will be integrated with constructive mission and engagement level effectiveness simulations during trade-off analyses to generate crew system requirements. Because CART is applying the same basic methodology for generating crew system requirements that is applied to generate system requirements, the method for generating system requirements will be reviewed briefly here.

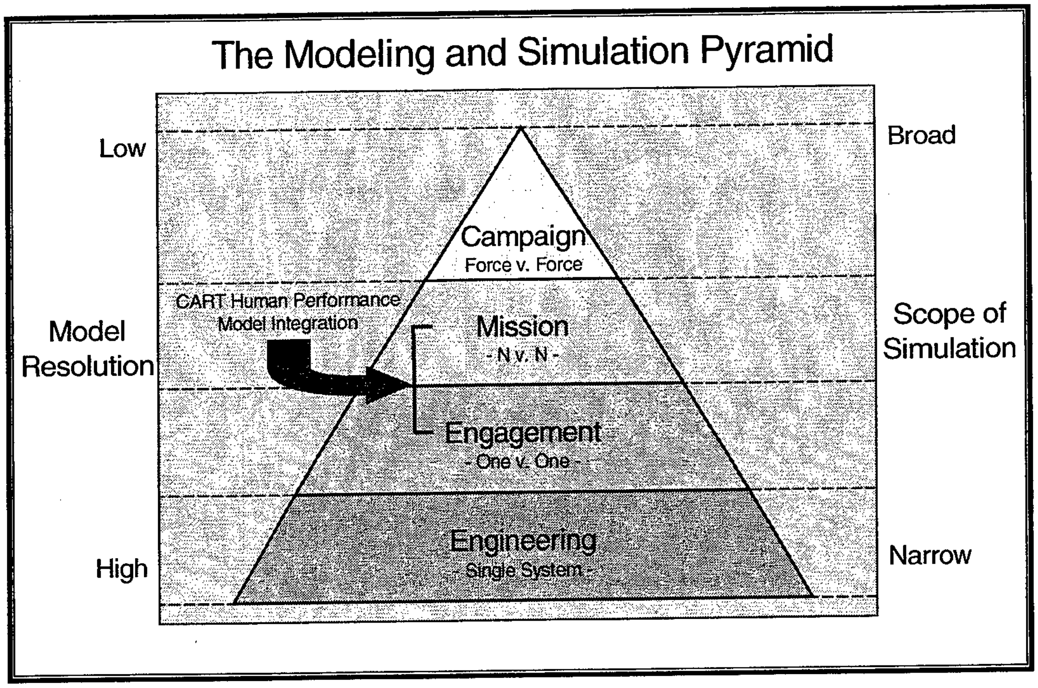


Figure 3. Levels of Effectiveness Modeling

3.3 Generating Subsystem Requirements with Modeling and Simulation

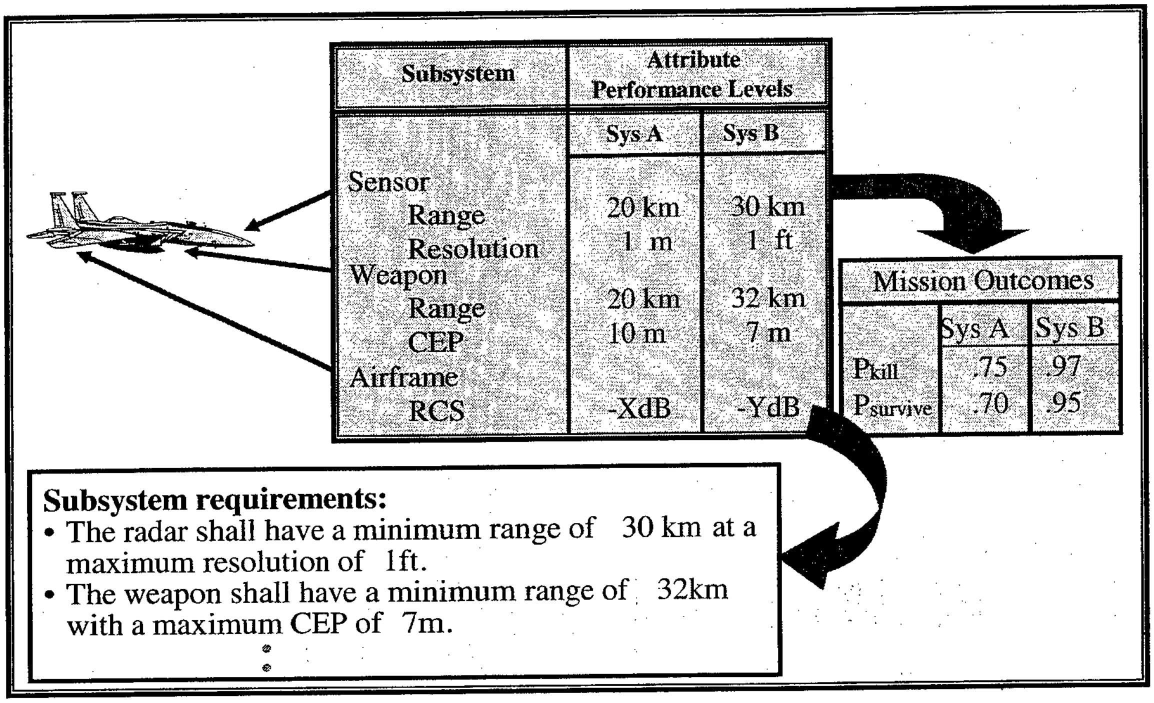
The first step to conducting simulation-based trade studies is to develop a simulation testbed that will support test activities. By testbed, we mean a simulation environment that has all the components necessary to exercise the system of interest in a representative mission context. The requirements for the testbed are derived from questions to be answered by the trade study. A study plan will be prepared that specifies the scenarios to be used, key characteristics and capabilities of the system to be tested, and measures of performance and effectiveness to be used to assess outcomes. In developing the testbed, attention is first directed at developing a simulation of the system. This is accomplished by combining models of key subsystems. For each subsystem, models are selected that provide the ability to manipulate important attributes of system performance specified in the study plan. For example, combining models for key tactical aircraft subsystems could create a constructive model of an advanced strike fighter. Models for subsystems such as sensors, weapons, and the airframe would be included. For the sensor model, the ability to manipulate attributes such as range and resolution would be provided. For the weapon, range and accuracy might be attributes that can be manipulated. For the airframe, it is the radar cross section (RCS). Within current constructive environments used for simulation-based acquisition, the primary focus is on representing physical components of the system, and the operator is ignored or treated in a very limited fashion.

A simulation of a system is integrated with a constructive mission environment to create a complete testbed. The constructive mission environment consists of entities with which the system under test interacts in the real world as defined by the scenarios from the test plan. Entities are selected that can have a significant impact on mission performance. The constructive system is exercised in the constructive mission environment and outcomes on mission performance are observed. For example, the constructive mission environment for a strike fighter testbed might include terrain, targets, threats, weather, and other models such as communications and command and control.

System requirements are derived from the results of testing the constructive system in the constructive mission environment. Figure 4 depicts the process employed. Within the constructive system models, levels of performance on key subsystem attributes are selectively varied in accordance with test matrices from the test plan. The constructive system is then employed in the constructive mission environment, performance data are collected, and mission outcomes are measured and analyzed to identify the levels of subsystem attribute performance that yield desired levels of mission performance. These subsystem attribute performance levels provide the basis for statements of system requirements, objectively and quantitatively stating levels of performance the subsystem must be able to achieve to satisfy the user's overarching mission requirements documented in the ORD and RCM.

In addition to being stated in objective, quantitative terms, a distinguishing feature of the requirements illustrated in Figure 4 is that they have a known, demonstrated relationship to mission performance. Because the requirements are based on an explicit linkage between subsystem and mission performance, there is a high degree of assurance that the user's desired mission requirements will be met. Thus, these requirements also can be characterized as *performance-based.*

In summary, development of objective, performance-based system requirements is possible because constructive simulations enable analysts to represent system alternatives and predict the mission performance they will generate. It is this predictive capability of simulation that makes the requirements derived from it so powerful, because each of the requirements has an explicit link to desired mission performance.



Subsystem	Attribute Performance Levels	
	Sys A	Sys B
Sensor		
Range	20 km	30 km
Resolution	1 m	1 ft
Weapon		
Range	20 km	32 km
CEP	10 m	7 m
Airframe		
RCS	-XdB	-YdB

Mission Outcomes		
	Sys A	Sys B
P_{kill}	.75	.97
$P_{survive}$	.70	.95

Subsystem requirements:

- The radar shall have a minimum range of 30 km at a maximum resolution of 1ft.
- The weapon shall have a minimum range of 32km with a maximum CEP of 7m.

⋮

Figure 4. How Subsystem Requirements Are Derived

3.4 The Lack of Operator Models and Its Consequences

Unfortunately, the models currently employed by acquisition analysts do not provide the ability to represent and manipulate operator performance in support of developing overall system requirements, or as needed to develop performance-based crew system requirements. Current constructive modeling environments (e.g., Suppressor, SWEG, and Brawler) are limited in terms of the range and type of operator activities that can be represented and manipulated. Suppressor, for example, has a 'Behavior Model' component that permits the user to apply a wide range of human behaviors to represent operator performance in tactical fighter platforms. The basic behavior set is, however, somewhat fixed and not readily extendable, and the ability to precisely control key performance attributes (e.g., time and accuracy) is limited (Pope, 1997). Another limitation of current models is the extent to which execution under specific conditions can be traced and understood. Brawler, for example, is a very detailed model in terms of representing what a fighter pilot might do in air-to-air combat. On a given run of Brawler, however, it is difficult to trace the execution of model components and understand why the components executed as they did. This hinders the ability to account for results, i.e. understanding the effect of pilot actions and tactics on results of the simulation.

Because acquisition analysts have not been able to model the operator effectively, the crew system has not been considered during the constructive-simulation-based trade studies conducted early in system acquisition. Thus, the crew system is omitted from the trade-off process that produces requirements for many other critical subsystems. Allocation of functions and tasks to the operator often is an implicit part of the early system concept definition and trade studies. Unfortunately, the impact of allocations on operator and mission performance cannot be evaluated without effectiveness models of operator performance. This inability to evaluate impacts of operator task allocations can lead to acceptance of system concepts that place unreasonable demands on the operator that cannot be mitigated by even the most effective crew system design.

4.0 HUMAN PERFORMANCE MODELING IN CART

4.1 Overview

CART will extend the current constructive methods used to generate system requirements to generation of performance-based crew system requirements by integrating human performance models with the constructive system simulations used today. This integration of a human performance model with a constructive system model is a critical element of the CART concept. Flach and Dominguez (1995) stress the need to jointly consider the operator and the system in the design process. Central to their approach to system design is the concept of constraints and boundaries. Put very simply, constraints are limits on performance that are inherent in the operator or system. Operators can be constrained in terms of the number and types of activities they can perform in a system (as determined by a task allocation scheme), as well as the speed and accuracy with which those activities can be performed (determined by factors such as physiology, training, and stressors). Systems are constrained by the physical capabilities of their subsystems. Attack aircraft systems, for example, are constrained in terms of factors such as speed, maneuverability, radar cross section, sensor type, sensor range and resolution, weapon type, and weapon range and accuracy.

Boundaries are performance levels associated with successful performance (or failure when exceeded). Consider, for example, an attack aircraft attempting to acquire a target with an infrared sensor. Assume that the weapon has a minimum engagement range of three miles. As such, acquisition must be complete within three miles of the target so that the target can be handed-off to a weapon and engaged. If the sensor's range is 12 miles, and the aircraft flies directly toward the target at 450 knots, the operator has approximately 70 seconds to complete the acquisition task. Thus, if the target is not acquired within 70 seconds, the operator will fail. In this scenario, the maximum sensor range and the minimum range at which hand-off can occur after acquisition defines the 70-second performance boundary. A change to either of these system performance constraints will change the performance boundary afforded to the operator. The mission environment also defines performance boundaries. For example, characteristics of a surface-to-air missile (e.g., range, speed, and maneuverability) define performance boundaries (e.g., reaction times, turn rates) within which the operator and system must perform to successfully evade the missile.

Given the above, we can say that a system will perform successfully if its operator and subsystem constraints do not exceed performance boundaries on key mission tasks. The challenge for system designers is to provide a design in which constraints operate within boundaries. The definition of -- and relationship between -- constraints and boundaries is complex and interdependent. In the evaluation of alternative system concepts, modeling and simulation provides a means for representing constraints and allowing them to interact with boundaries that naturally result within the system and mission environment. When the operator is not represented in these simulations, an important source of constraints is not considered. Conversely, when operator models are run as a stand-alone simulation (as is done today) representation of system and mission performance boundaries on operator performance is omitted.

The value of having a holistic constructive testbed that integrates the operator, the system, and the mission environment is that it permits analysts to explore the complete constraint space associated with alternative system concepts and to vary the boundaries associated with each. This is particularly valuable for the crew system designer. Constructive system and mission models provide an opportunity to represent these boundaries realistically and to demonstrate changes in the boundaries as a function of changes in system alternatives (including different levels of automation and operator abilities).

4.2 Human as an Information Processor

As noted above, specification of performance-based crew system requirements is based on a demonstrated relationship between operator performance and a desired level of mission performance. In CART, human performance models will be used to establish that relationship. The human performance models will enable analysts to represent operator constraints and have them interact with performance boundaries that exist in the system and mission environment. A critical feature of the human performance models will be the representation of operator constraints. Two constraints have already been discussed that relate directly to crew system requirements. These are the tasks assigned to an operator and the levels of performance associated with those tasks. Thus, the human performance model must be capable of representing tasks assigned to an operator with different system alternatives and varying performance on the critical attributes of performance associated with those tasks. This requires a significant degree of flexibility in human performance modeling capability because operator tasks vary from one mission context to another (e.g., tactical fighter pilot vs. an airborne warning and control officer), and the attributes of performance that are important vary as well.

A particularly important, though broader, constraint is the conceptual basis used for representing human performance. This will affect how the tasks are allocated to the operator and are organized and controlled within the model. Pew and Mavor (1998) note the need for "an integrative model that subsumes all or most of the contributors to human performance capacities and limitations." Pew and Mavor point out that most integrative architectures used today view the human as an information processor. CART has adopted this framework for representing human performance.

The basic concept of the human as an information processor (HIP) model is that the operator adapts and organizes tasks to meet current demands of the mission. Though a task allocation scheme defines the potential set of activities the operator can perform, the information processor model determines which task gets performed at a given time. Adapted from Hendy and Farrell (1997), Figure 5 illustrates the HIP model.

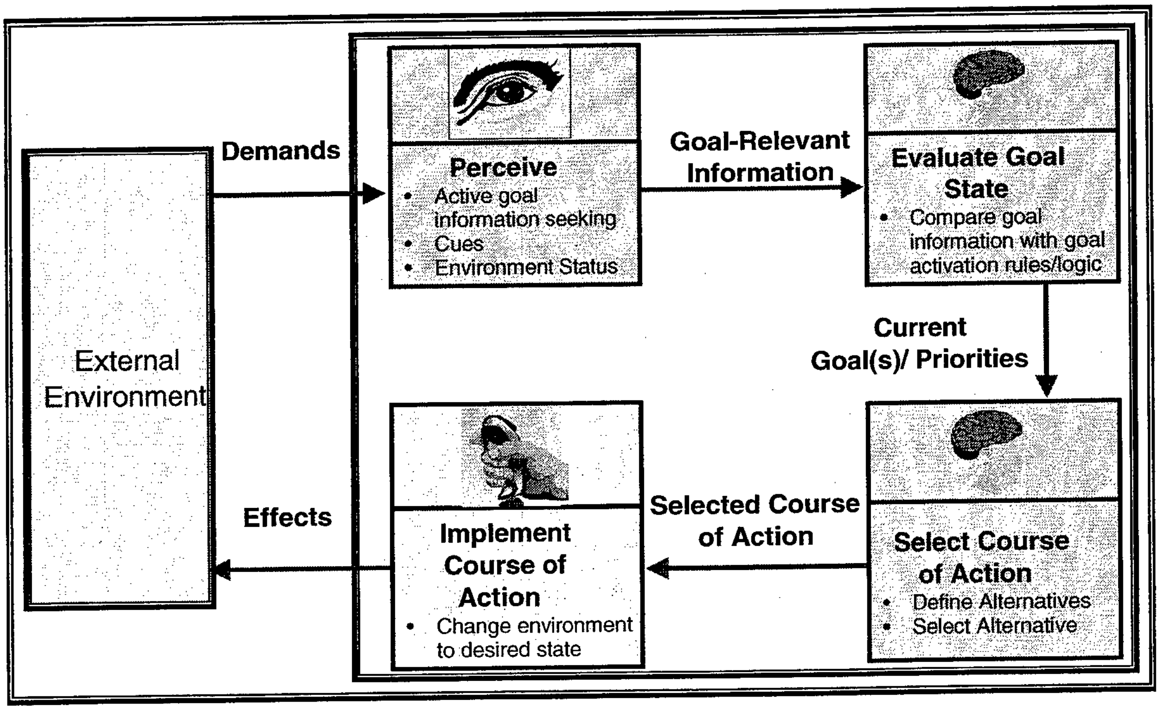


Figure 5. Basic Human Information Processor (HIP) Model

Within the HIP elements depicted in Figure 5, the central construct is the notion of goals. Given that multiple mission demands can be active at the same time, a mechanism is needed to sort among concurrent demands to choose which goal(s) get serviced first. The model assumes that in a given system-mission environment the operator has an internal goal structure that helps assess and prioritize demands to be met. These goals are associated with functions that must be performed successfully to accomplish the mission. The goals are defined in terms of states of the external environment that the operator seeks to control. In goal state evaluation, information from the environment provided by perceptual processes is compared with internally held knowledge of expectations about world states and rules for when goals become active. When conditions expressed in goal rules are met, the goal state becomes active.

Goals need information from the environment to determine when they become active. This information is obtained via operator perceptual processes, using the five senses. The 'Perceiving' block in Figure 5 illustrates how these perceptual actions feed goals. Perception of demands is an active process in which the operator purposefully seeks specific information required by the particular goal set that is driving operator performance. Representation of this perceptual activity in a human performance model is important because such activity often is not identified explicitly in task allocation schemes. Thus, it becomes an additional set of performances that can, in turn, constrain the core set of operator activities defined in task allocation, i.e., the operator can only perform those tasks that need attention in which goal-relevant information has first been perceived.

When goals become active, attention turns to selecting a course-of-action for bringing the current state of the world into the desired state. Within a system there may be a number of capabilities that can be applied to achieve a goal. In a given situation, some might be more effective than others. Course-of-action selection involves selecting the capabilities and methods for implementation that best accomplish the goal. The emphasis of course-of-action selection is on decision making. Course-of-action selection can involve a variety of cognitive processing skills (e.g., skill-based, rule-based, knowledge-based reasoning). It can also involve other perceptual and action components that are applied to gain additional information needed to select a course-of-action. Again, this is a dimension of human performance not often addressed during the task allocation phase of system design. Indeed, the specific requirements for course-of-action selection can be influenced greatly by the configuration of a system defined in an alternative (e.g., aircraft with on-board mission planners can significantly reduce pilot route re-planning requirements while in flight). Like the perceptual activity described above, course-of-action selection is another overhead activity associated with the operator's management of his performance.

Once a course-of-action is selected, it is implemented. Course-of-action implementation generally involves motor activity (e.g., manipulate a control or throw a switch), though when implementation activities are complex, perceptual and cognitive activities might be involved to control and manage course-of-action implementation, dependent upon specific environmental conditions. The objective of course-of-action implementation is to produce an effect desired by a goal state on the environment (e.g., attack actions seek to destroy a target, evasion actions seek to evade a threat). Observation of effects is performed by perceptual capabilities, which, in turn, drive the goal states. The cycle repeats itself until the desired state is achieved. Readers familiar with control theory will recognize that the information processor model is really a type of closed-loop control model (Flach, 1990).

Finally, another important human performance constraint that must be represented by human performance models is the limitation of perceptual, cognitive, and motor resources in terms of the number of concurrent activities that can be supported. As noted earlier, operators in complex modern military systems will have multiple, concurrent demands. As such, there will also be multiple active goal states to manage simultaneously. Active goals are dynamic and will shift in response to changing conditions in the mission environment. Activities under active goals often compete for the same human performance resources (perceptual, cognitive, and motor). When resources are exceeded by demands, excessive workload will result, and the operator will then engage in workload mitigation strategies to manage the demand (Hendy and Farrell, 1997). The operator might suspend or completely shed lower priority activities. Alternatively, the operator might choose to work two concurrent activities simultaneously with the result that performance time for both activities is extended significantly as resources are shared between the two. Finally, the operator might employ a less effective but more efficient processing solution for an activity. In the process of applying these different workload mitigation strategies, some mission demands might not be met at all, others might not be met within the required time window, and still others might not be met because some other dimension of task performance (e.g., accuracy) is compromised. The net result of all possible workload effects, however, is that mission performance can suffer. This result will be reflected as a consequence of properly representing the number of concurrent activities that can be supported in the human performance model.

4.3 CART Human Performance Modeling Architecture

CART's architecture for integrating human performance models into engagement level simulations is presented in Figure 6. The design of this architecture was driven by three broad sets of requirements: 1) the previously-described human performance modeling capabilities needed to generate performance-based crew system requirements, 2) the need to integrate the CART human performance modeling capability with a variety of current and future constructive simulation testbeds composed of a wide range of models and simulations, and 3) the need for a relatively easy-to-use tool that can be applied by the personnel that construct, maintain, modify, and operate those constructive testbeds. The following discussion describes how each of these requirements is met in the CART architecture.

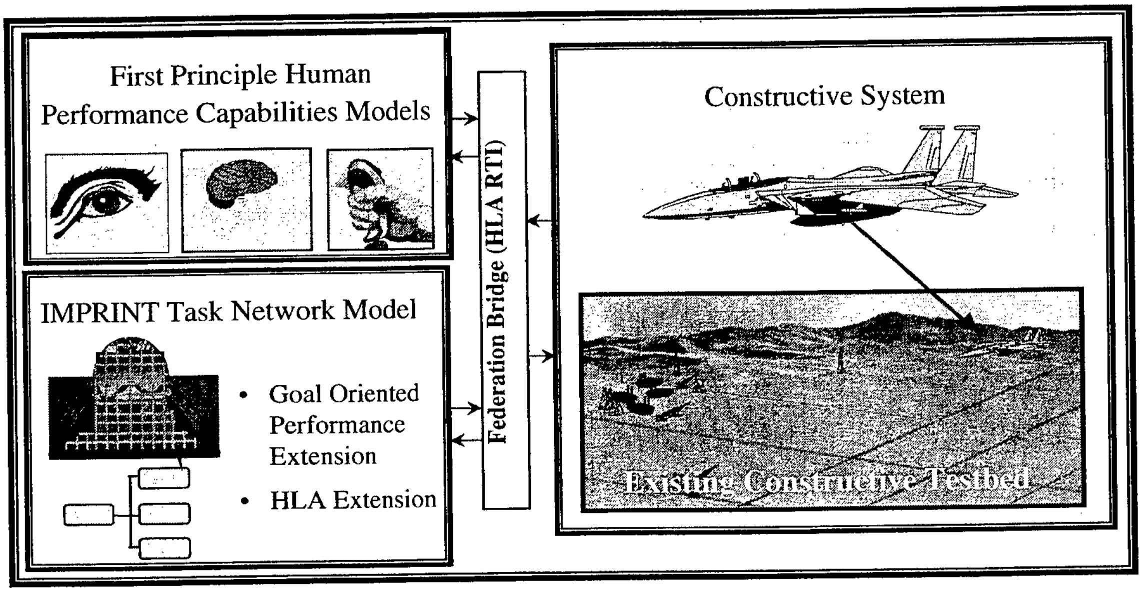


Figure 6. CART Human Performance Modeling Architecture

4.3.1 Human Performance Modeling Capabilities for Generating Performance-Based Crew System Requirements

As indicated in Figure 6, CART employs a hybrid architecture (Pew and Mavor, 1998) for modeling human performance. Task network modeling will be the core human performance modeling method. The Improved Performance Research Integration Tool (IMPRINT) will be the particular task network modeling capability employed. IMPRINT was developed by the Army

Research Laboratory to investigate manpower, personnel, and training issues in Army systems. It provides a near off-the-shelf, mature solution to CART human performance modeling needs that leverages a significant investment by the Government.

With regard to the objective of generating performance-based crew system requirements, IMPRINT is more than a model of human performance; it is a task network modeling environment that can be used to create human performance models that are tailored to specific operator performance requirements in a system. Within IMPRINT, operator performance dictated by task allocation schemes is broken down into a series of tasks characterized in terms of performance times, accuracy, and probabilities. Thus, IMPRINT meets the requirements to model a range of different operators and tasks and to be able to manipulate critical dimensions of operator performance. Though the HIP is not implemented explicitly in IMPRINT, the general features and capabilities of task network modeling can be combined and applied to represent a HIP model (Hendy and Farrell, 1997). In addition, CART is extending IMPRINT capabilities to incorporate a new feature called 'goal states'. Goal states are specialized functions that will enable analysts to specify goals that will drive operator model performance, specify trigger conditions to goals, and have task networks associated with the goals that conduct course-of-action selection and implementation. Thus, a more explicit representation of the HIP will be provided by CART versus that currently represented in IMPRINT. Finally, a variety of models for implementing workload effects in task network models have been implemented or proposed (Farmer et al. 1995; IMPRINT Users Guide, 1998; Hendy and Farrell, 1997), thereby supporting the requirements to represent workload in the HIP.

While task network models provide an easily-understood representation of human performance, their fidelity is limited in terms of modeling in detail specific human capabilities such as cognition and perception. For this reason, CART users will have the ability to augment the task network models with *first-principle models*, which provide high fidelity representations of human capabilities. These first-principle models may come from more detailed, sophisticated human performance models and modeling tools such as Soar, OMAR, or COGNET (see Pew and Mavor, 1998, for a review of these and other integrative architectures). The basic requirements for first-principle models in CART are that they be able to communicate via the High-Level Architecture (HLA) (see the following paragraph) and that they provide data needed by the task network model. Essentially, tasks in the task network will call first-principle models that represent the capabilities required in a task. A first-principle model will execute a particular human performance capability (e.g., perception) and return the parameters required by the task network model.

4.3.2 Integration of CART Human Performance Models with Different Constructive Testbeds

The Defense Modeling and Simulation Office's (DMSO) High Level Architecture (HLA) will provide the communications link between models. In CART, data will be passed between architecture components using the HLA Run Time Infrastructure (RTI). The task network model will receive data about system and mission status from the constructive system simulation and data about the external world (e.g., SAM launches) from the mission environment models via the RTI. Actions to be implemented by the system (e.g., maneuver, target designation, weapon launch) will be passed to the constructive simulation by the task network model via the RTI. Similarly, the task network model will use the RTI to pass data to first-principle models to initiate them and receive the results of first-principle model execution. HLA was selected as the communication interface because it is the Department of Defense-wide solution for providing a common technical framework for modeling and simulation (DOD 5000.59-P, 1995). It can be expected that most simulations in the future will have the capability to interface via the HLA.

4.3.3 An Easy to Use Tool that Can Be Applied by Personnel Who Support Constructive Testbeds

It is expected that CART will be implemented by the team that develops, maintains, and operates the constructive simulation testbed within which CART will be integrated. This team will be made up of members with expertise in operations research and analysis, human factors, test design, weapon system operations, simulation, and modeling. Though human factors expertise is expected on the team, it is unlikely that these personnel will be trained, experienced modelers or advanced behavioral theorists. CART's modeling capability must make model development a user-friendly process that can be applied by relatively inexperienced personnel with a basic human factors background. As a mature, stable human performance-modeling tool, IMPRINT has demonstrated its ability to effectively support novice users. Within CART, IMPRINT's usability will be enhanced by providing a library of operator models that Air Force users can apply to reduce the effort required for developing their own models prior to conducting robust trade-off analyses.

4.4 Using Task Networks to Implement the Human Information Processor Model

As noted above, the HIP model is not implemented explicitly in IMPRINT. Representation of the HIP is achieved by applying elemental functions and capabilities of IMPRINT's task network modeling capability to produce the characteristics of a HIP. Because the HIP is a central construct within CART, its implementation using task network modeling bears some discussion. This section describes the basic components of task network modeling that will be used to create the HIP and then describes how the components will be integrated to yield a functioning HIP. The discussion on task network modeling centers on four components: goal functions, tasks, task networks, and variables and macros.

4.4.1 Goal Functions

In task network modeling, operator activities are organized into functions and tasks. Generally, functions provide a means of organizing tasks at a high level but have no direct control over execution of tasks in the model. The CART program is providing an important extension to IMPRINT functions called 'goal functions'. Goal functions will allow users to represent the operation of goal states as described in the HIP model, with task networks being associated with specific operator goals states. Once a goal states becomes active, the networks will perform activities that implement the goal. Thus, in CART, goal functions will have explicit control over the execution of tasks in the model.

Specification of goal states via CART goal functions is a two step process. In the first step, the user defines the goals to be used in a model. For each goal the user specifies the name and a description, enters a numeric priority value (the lower the number, the higher the priority), and specifies the conditions under which the goal is initiated or triggered. Initiating conditions are entered as an expression based on user-defined variables or macros (discussed later in this report). As an example, a pilot of a tactical strike fighter might have goals of navigating to a target area, acquiring a target, and evading threats along the way. Associated with each goal state would be a network of tasks that work to achieve the goal. Of the three goal states, evading threats would be the highest priority goal, followed by target acquisition and navigation. Threat evasion might initiate when an enemy air defense system acquires the aircraft or when a missile is launched on the aircraft.

The second step in specifying goal function data defines the logic that controls how individual goals are executed when multiple goals are triggered at the same time. (In this step, goals are listed in priority order (highest to lowest). For each goal function, the user specifies the effect on any other lower priority goal functions that might be running when that goal function is triggered. The user can direct that a lower priority goal *Do Nothing*, *Suspend*, or *Abort*. *Do Nothing* allows the network associated with the lower priority goal function to continue to execute. *Suspend* halts execution of the network associated with the goal function, and if execution is resumed later, it will begin at the point it was executing when it was suspended. *Abort* also halts execution of the network associated with the goal function; however if execution of the network is resumed later, it will restart at the beginning of the network. The user can also specify where the network associated with the triggered goal begins execution. The options are *Resume* and *Restart*. *Resume* is used when the task network associated with the goal might have been halted previously and the user wishes to resume execution from the point at which it was halted. With *Restart*, execution of the network is restarted from the beginning of the network.

Continuing with the above example of the tactical strike fighter, a user would specify what happens to the target acquisition and navigation goal functions (lower priority goals) if they are active when the threat evasion goal state initiates. Because threat evasion is so important to survival, it is likely that a user would cease activity associated with any other goal functions. Target acquisition, if active, would be suspended. This assumes that target acquisition involves a sequence of target search activities. If the sequence has been initiated prior to the need to evade the threat, it would be desirable to pick up the sequence where it was suspended when the target acquisition goal function is resumed. The navigation goal function would likely be aborted. This would permit the network associated with navigation to start from the beginning when it triggers again. This assumes that the navigation task network is cyclical, involving assessment of current status with respect to a planned route and adjusting aircraft altitude, speed, and heading accordingly.

4.4.2 Tasks

In task network modeling, tasks are the lowest level of decomposition. The particular content and level of detail applied in a task is up to the user. For example, a task might be specified as *Image the Target with the Radar*. Alternatively, imaging the target with the radar might be represented as a series of tasks (e.g., *Select a List Point, Select Radar Mode, Select Radar Range, Energize Radar*). It is at the task level that the perceptual, cognitive, and motor activities involved in the information processor will be represented and manipulated. For individual tasks, a developer can specify a variety of data that are used to define key performance attributes, control execution and effects of the task. Data elements essential to CART are listed below.

Task Performance Time. The time taken to execute the task is specified as a mean time with an associated variance. The type of distribution associated with the variance can be specified as well. Mean time can be expressed as a fixed value or as an expression that varies mean time as a function of a given factor (e.g., stress, fatigue).

Accuracy. The user can specify a dimension along which accuracy of performance is important (e.g., pixel error in cursor placement) and then specify a mean accuracy and an associated variance. This is another component of performance attribute manipulation.

Release Conditions. If there are special conditions that determine when a task should begin executing, the user can establish release conditions for the task. Release conditions are specified in an expression using user-defined variables (discussed below). Release of a target engagement task, for example, requires that the target is in range of the weapon. A release condition for a target engagement task might be specified as *Target_range <= 3*, where *3* is the range of the weapon in miles and *Target_range* is a variable that reflects changing range to the target.

Effects. Effects provide the user with a means of changing states or conditions within a model. If, for example, fatigue is an effect that is being used in a model and current fatigue levels are being tracked in a variable called *Fatigue*, an ending effect of a very physical task might be to increment the fatigue variable by some value. Within CART, the effects fields of a task will provide the means for updating variables that pass actions to the constructive system simulation. Recall that IMPRINT human performance models will communicate with constructive system simulations via the HLA RTI. Communication between models involves passing data in the form of variables. If, for example, a human performance model needs to change the heading of the aircraft it is controlling, the HPM might update the value of a variable called *Heading*. The *Heading* variable would be passed to the constructive system simulation which would change aircraft heading to match the value contained in *Heading*. Within the HPM, updates to the variable called *Heading* would be made in the effects field of a task specified for evaluating and updating aircraft heading. An expression or series of expressions would be provided that contain algorithms for determining the heading required. The final portion of the expression might read something like *Heading := New_Heading*, where *New_Heading* is a variable containing the value of the current heading to be flown and *Heading* is the variable that passes the update to the constructive system simulation.

Workload. Within IMPRINT, tasks can be characterized in terms of their workload demands as measured by the Visual, Auditory, Cognitive, and Psychomotor (VACP) method (Aldrich et al. 1984). Within each category, the user is given a scale from which a rating is selected to characterize the task. Under the visual category, for example, a rating of 1.0 is the workload associated with a simple visual detection task. A rating of 7.0 indicates the workload for a visual task involving continuous scanning, searching, and monitoring. Under the cognitive category, a task involving simple associations gets a rating of 1.0 and a task involving estimation, calculation, and conversion receives a rating of 7.0. When human performance models run under IMPRINT, the VACP workload modeling module totals VACP ratings across all tasks that are running at any one point in time. Thus, a moment-to-moment total workload score is computed for each VACP category. Typically, these data are used as a basis for tracking momentary workload levels in a model. In CART it will be used as a basis for implementing effects of workload on mission performance and effectiveness. As an HPM executes, changes in total workload across the VACP categories will be monitored. When workload in a category reaches a predefined maximum, specialized task control functions in IMPRINT will be applied to suspend or abort performance of lower priority tasks whose workload would push the total workload values over the maximum limit allowed. Thus, tasks that need to occur will be delayed or will not occur at all. This can lead to mission performance failures (the effect of excessive workload).

4.4.3 Task Networks

Tasks are linked into networks that define paths and sequences of events. These networks can be used to represent different aspects of information processor performance. For example, networks can be used to represent perceptual activities involved in seeking information for goal states. Within this network, tasks would be specified that represent the selective attention to displays, instruments, etc. that occurs in a crew station. The sequence and duration of tasks would determine what the operator sees and when it is seen. Branching nodes are a feature of networks that can be used to increase model complexity by representing alternative paths and decision making. Branching nodes can be deterministic or probabilistic. Deterministic branching provides a means of representing operator reasoning, and it is based on logic provided by the user in expressions or macros (see below). Probabilistic branching can represent the inherent variability of the human operator. Multiple branches off a decision node can represent alternative courses of action. The networks that implement a fixed course-of-action represent procedural knowledge about how to accomplish a particular function.

4.4.4 Variables and Macros

Within the model, users can define and manipulate variables. User-defined variables provide a means of storing information that changes over the course of model execution and that is used by the model to influence goal and task execution. Returning to the example of the fatigue variable described above, this variable was used to track changes in physical fatigue as a function of tasks performed in the model. The fatigue variables' value could be changed (using the effects field) as tasks with physical components were performed. This information on fatigue could, in turn, be used by the model to moderate execution of tasks. For physical tasks, task duration could be defined in an expression in which fatigue is a variable. Thus, when fatigue levels are higher, task duration is longer. The result is a model that adjusts performance based on changing conditions in the simulation and, consequently, provides a more realistic representation of the dynamics of human performance. In terms of human performance capabilities, user-defined variables provide an ability to represent short-term memory and declarative knowledge. Section 4.4.5 illustrates how user defined variables are used to represent short-term memory.

As indicated in the discussion of effects fields, a particularly important application of user-defined variables in CART will be to provide communications with the constructive system simulation. The HLA interface between the CART human performance model and the constructive simulation will provide a communications link between the two environments. Data coming into the human performance model will support information (perceptual) needs of the model, while data being passed to the constructive system simulation will direct actions that the constructive simulation must perform. As part of an HLA implementation, simulation object models (SOMs) and federation object models (FOMs) will be developed that define in detail the data interchange that occurs between models. As part of the HLA extensions being provided for IMPRINT, users will be able to define variables within the human performance model that are linked to the data being passed across the RTI. The CART RTI interface software will manage data transfer between the RTI and user-defined variables. Two basic operations will be performed at run-time: (1) user-defined variables that receive information from the constructive simulation will be updated as that information changes, and (2) values of variables that direct actions of the constructive simulation will be extracted and passed back to the constructive simulation.

Finally, macros are a specialized capability that performs calculations and logical operations. They are, in effect, a mini-program. The value of macros is that they provide the ability to represent fairly simple (rule-based) reasoning and decision making. When combined with network branching capabilities, macros can be used to represent complex decision making.

4.4.5 Implementing the Human Information Processor

Figure 7 depicts how the HIP model will be implemented within CART using task network modeling. Components of the 'HIP Model' (from Figure 5) are depicted at the top of the figure for reference. HIP 'Perceive' is implemented in the 'Perceptual Tasks' and 'Short Term Memory' blocks of the 'Human Performance Model'. 'Evaluate Goal State' is implemented in the 'Goal Functions' blocks. 'Select Course of Action' and 'Implement Course of Action' are implemented in the 'Sub Net' decision nodes and the network of tasks associated with the selected course of action, respectively. The following paragraphs step through this implementation in more detail.

The story told in Figure 7 begins with the 'Constructive Simulation Environment' depicted in the upper left portion of the figure. The 'Constructive Simulation Environment' provides the representation of demands to the 'Human Performance Model'. The HLA RTI controls passage of data (represented in the box labeled 'RTI Data') between the human performance model and the constructive testbed.

Within the 'Human Performance Model', a network of tasks (labeled 'Perceptual Tasks' in the figure) will perform 'information seeking' about current demands in the mission environment. The perceptual task network represents the sequence and timing according to which the operator model 'observes' displays and instruments and 'listens' for communications, tones, alarms, etc. Perceptual information seeking is driven by information needs for evaluating goals. In the HIP model, perceptual tasks enable the user to build a mental model of the current situation in the mission. This mental model is retained in short-term memory where it is updated as the perceptual tasks are repeated. In CART, user-defined variables will be used to represent short-term memory. Within each perceptual task in the perceptual task network, an expression will be provided in its effects field that updates the user-defined variable(s) that represents the short-term memory component associated with the perceptual task. In the figure for example, there is a perceptual task called *Check Speed.* When this task executes, it reads data from a variable at the RTI called *Speed* and updates data in a user defined variable called *Airspeed,* using the simple expression *Airspeed := Speed* in the task's effects field. Thus, the human performance model's mental model of speed is updated only when the *Check Speed* task is performed. This reflects the fact that perception is a constrained process. We cannot know everything about our world instantaneously. What we know is determined by when it is perceived which is, in turn, controlled by our 'schedule' of perceptual activity.

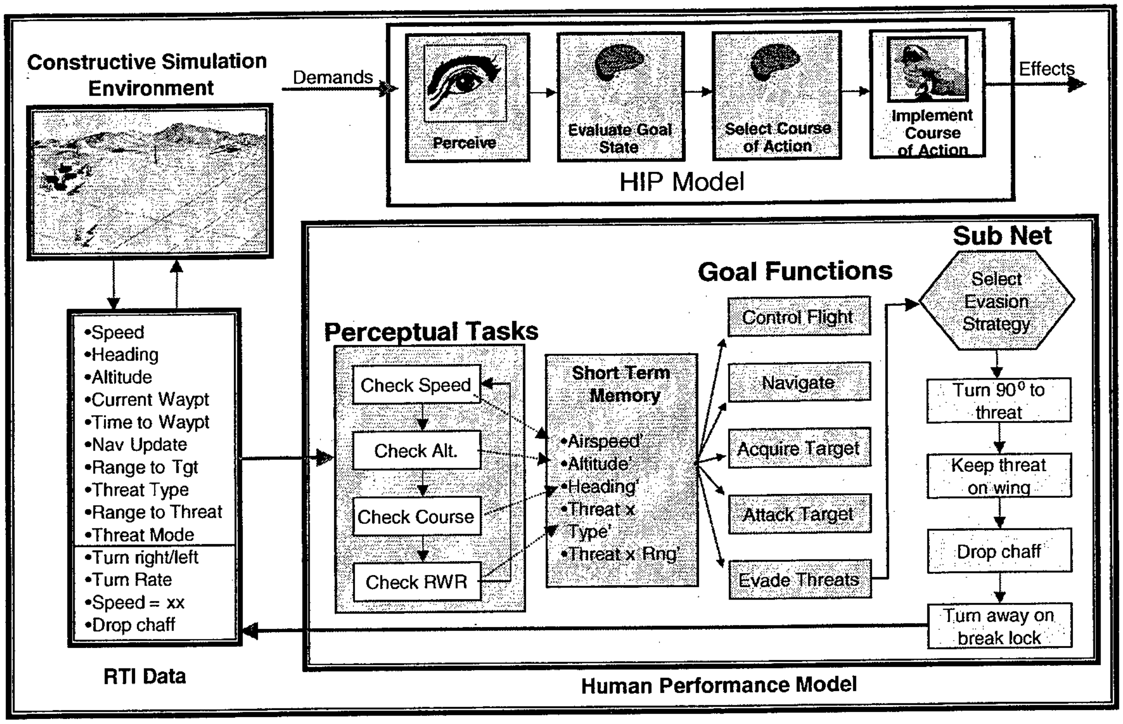


Figure 7. Implementation of the HIP Using Task Network Modeling

In the HIP, goal states are evaluated based upon the contents of short-term memory. Thus, conditions in the environment can change such that a goal should become active, but the goal will not actually trigger until that new condition is perceived and reflected in short term memory. As described above, this same process is represented in the IMPRINT human performance model.

Within the human performance model, goal functions evaluate on every cycle of the model. Initiating condition expressions provided by the user determine when a goal triggers. These initiating condition expressions evaluate mission environment conditions represented by the user-defined variables that reflect short-term memory. Once initiating conditions have been met, additional logic provided by the user on goal priority and activation in relation to other higher priority goals determines whether the goal actually becomes active.

For example, the goal function *Evade Threats* would trigger when threats are present within a certain proximity to the aircraft. The radar warning receiver (RWR) would be the aircraft subsystem that provides threat data to the operator. The operator model would perceive data from the RTI associated with the RWR (e.g., threat type and range). Because threat evasion is such a high priority goal, the model developer may decide to suspend activity under any other goal state that might be triggered while the evasion goal state is active.

For goals that become active, task sub-networks are activated. Within these sub-networks, decision nodes attached to branches can represent alternative courses of action. Within the decision node, logic can be specified that selects the course-of-action best-suited for the current circumstances. When decision making is complex, involving a series of decisions, course-of-action selection itself might be represented by a network of tasks. Under our threat evasion example, selecting a course-of-action would probably involve considering the type of threat and choosing from among a set of evasion options (e.g., maneuver, countermeasures, or a mix of both).

Each branch off a course-of-action selection task represents an alternative course-of-action -- of which, only one will be selected and executed. While the emphasis is on action, tasks within the network can be devoted to perceiving additional information from the environment needed for action implementation and reasoning needed to moderate or control action implementation. As the course-of-action executes, inputs to the constructive system are provided via updates to user-defined variables that, in turn, update variables in the RTI. The constructive system model receives these data from the RTI and then changes its performance accordingly. Continuing with our threat evasion example, a course-of-action implementation strategy might be applied that involves maneuvering the aircraft while applying some countermeasures. As the task network model executes these actions, data are sent across the RTI to command the constructive system models to implement the corresponding actions.

5.0 CART IMPLEMENTATION

The discussion thus far has focused on CART's application of human performance modeling and its integration with the system acquisition process via today's Simulation-Based Acquisition (SBA) process. However, CART is *more* than a human performance modeling technology. It is also a process for applying that technology in conjunction with other system modeling and evaluation activities to generate system and subsystem (including crew system) requirements. In this section, we outline a number of activities that will be involved in implementing CART within an acquisition program. To better illustrate the process and products, we describe them within the context of a hypothetical strike fighter acquisition program. In these examples, it is assumed that the acquisition program is examining alternative concepts for an air-to-ground targeting radar. The discussion that follows is organized into two major subsections. The first discusses important factors and considerations that will drive CART implementation. The second section describes the implementation and application of CART technology.

5.1 Factors and Considerations that Drive CART Implementation

Before initiating CART implementation activities, it is essential to fully understand the plan for conducting trade studies associated with the system to be acquired. In particular, there are four elements of the study plan that will drive CART implementation (see Figure 8). These are the scenarios to be used in the test, the alternative system concepts to be tested, measures of effectiveness and performance to be applied to evaluate alternative system performance, and the simulation environment to be used to represent the system and mission environment. The test scenarios describe the operational environment in which the system will be employed. They provide insight into the kinds of entities with which the system interacts during a mission, the system concept of operations, and the tactics employed. Understanding the test scenarios is a critical requirement for building effective simulations of the mission environment. In our example of a hypothetical strike fighter acquisition program, a scenario might be specified that has the aircraft ingress to the target area at an altitude of 15,000 ft., routing around threats to avoid them, and searching for a Theater Ballistic Missile whose location is known only approximately. The scenario might also dictate that the strike fighter will receive updates to target location via a digital data link from an off-board source. Given the scenario, we can identify critical features of the mission environment that must be represented in a simulation testbed (e.g., threats, targets).

The description of system alternatives to be evaluated details the capabilities of each alternative. Alternative descriptions are important because they provide insight into system capabilities that must be modeled and with which the operator model will interact. In our hypothetical strike fighter example, two alternative air-to-ground targeting radar concepts are being evaluated. One has a 20 nm range, a resolution of 1 m, and a display area of 6” by 6” (480 pixels x 480 pixels). The second alternative has a 30 nm range, a resolution of 1 ft, and a display area of 12” by 12” (960 pixels x 960 pixels). Characteristics of these alternatives have implications for operator performance. Operator performance can be affected in terms of number of images required to search for a target and the level of detail associated with a target image.

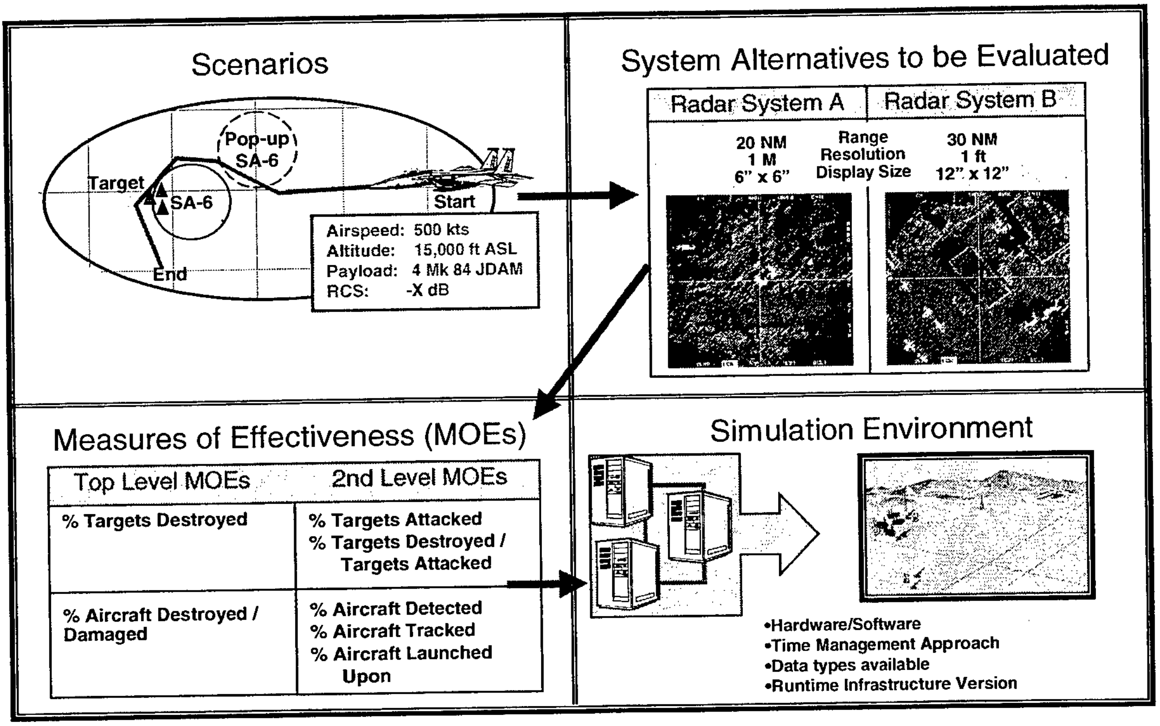


Figure 8. Factors and Considerations that Drive CART Implementation

The third element of the study plan that must be fully understood prior to CART implementation consists of the metrics upon which system performance is evaluated. These measures of effectiveness (MOEs) are developed by the acquisition program and are used as metrics in the AoA process during the Concept Exploration phase of acquisition. Often, the MOEs are at a very high level, representing *operational* measures of mission effectiveness and system performance being assessed and predicted through the modeling and simulation activities. Such high-level measures may include ‘percent targets destroyed’ or ‘percent aircraft destroyed / damaged’. Later in the CART implementation process, these MOEs will be expanded to a lower level to allow a more detailed understanding of operator task and subtask performance impacts on overall mission effectiveness.

Finally, the program’s constructive simulation environment must also be well understood. Going beyond the simulation scenarios, issues such as simulation system hardware and software, time management, data available in the simulation and federation object models, and version of the runtime infrastructure must be addressed in order to develop and integrate a fully compatible human performance model.

5.2 Overview of CART Implementation Activities

Once the up front familiarization with the program is complete, the CART implementation process can begin. The CART process for integrating and utilizing human performance modeling within the constructive simulation environment involves four basic steps, shown in Figure 9. First, for each system under consideration, a series of mission decomposition activities is performed in an effort to fully understand the various human and system tasks performed during the mission. Next, using the modified IMPRINT task network modeling tool described earlier, human performance models are developed that will subsequently be integrated into the acquisition program’s constructive simulation environment. Simulation trials are then run and data analyzed to identify levels of task performance that are key drivers in determining mission success. Finally, these critical levels of task performance are translated into crew system requirements that identify operator levels of performance that the crew system must support in order to achieve desired mission outcomes. The following discussion provides a more detailed explanation of these activities as well as examples of the products from each.

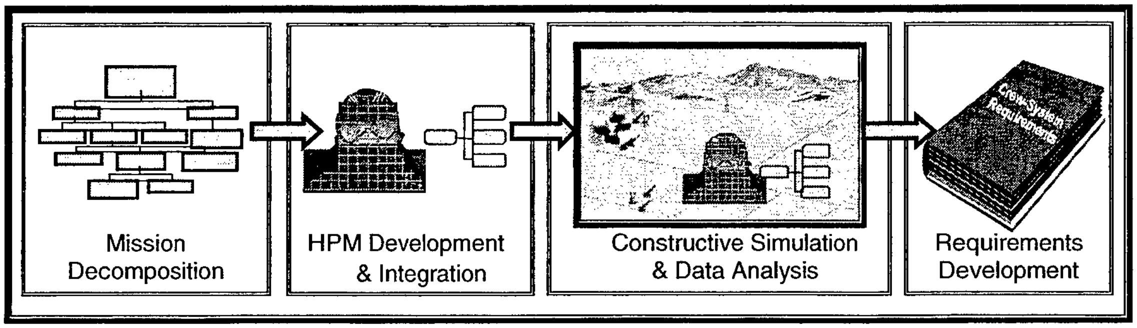


Figure 9. The CART Implementation Process

5.2.1 Mission Decomposition

Perhaps the most critical activities to be performed in the CART implementation process will be associated with fully decomposing the system's mission to gain a detailed understanding of the various the tasks performed. A series of decomposition activities will be conducted in an effort to identify key operator tasks that need to be represented in the human performance models and to identify how performance on those tasks relates to overall mission outcomes. These activities include a means-ends decomposition, specification of goal state definition and interaction, information-decision-action analysis, and MOE extension and data collection definition. Each is described below.

Means-Ends Decomposition. Based loosely on the "abstraction hierarchy" put forth by Rasmussen, Pejterson, and Goodstein (1994), the means-ends decomposition is structured in a hierarchy that represents various attributes of the system, from its high-level purpose down to the physical components required to perform any actions. To develop this decomposition, subject matter experts (e.g., experienced strike fighter pilots) are interviewed to identify the mission purpose, as well as the functions and corresponding goals, tasks, and subtasks required to perform the mission. At the lowest level, these experts also identify the physical equipment with which the task or subtask is performed.

Figure 10 shows a portion of a means-ends decomposition for a strike mission against a fixed facility target. The decomposition begins at the mission level, specifying the primary mission purpose. In this case the mission purpose is to destroy critical enemy ground targets. In support of this mission, the system must perform four general functions including, flight control & navigation, threat avoidance, target acquisition, and attack. These functions also correspond to high-level pilot goal states that will be incorporated into the human performance models. Each general function is achieved through a series of tasks. In this example, the *Acquire Target* function is comprised of a target aimpoint selection task, a target imaging task, and an aimpoint designation and acceptance task. Subsequently, each of these tasks is broken down into its constituent sub-tasks. Designation and acceptance of the aimpoint consists of locating the desired aimpoint on the image, slewing the cursor to the desired pixel on the image, and designating the aimpoint to update the target coordinates. Finally, the physical activity in the task or subtask is then further decomposed into the physical form and configuration associated with its performance. For the *Slew Cursor to Desired Pixel* task, two key hardware components are deemed necessary: the radar display that shows the radar image and the control mechanism used to manipulate the cursor.

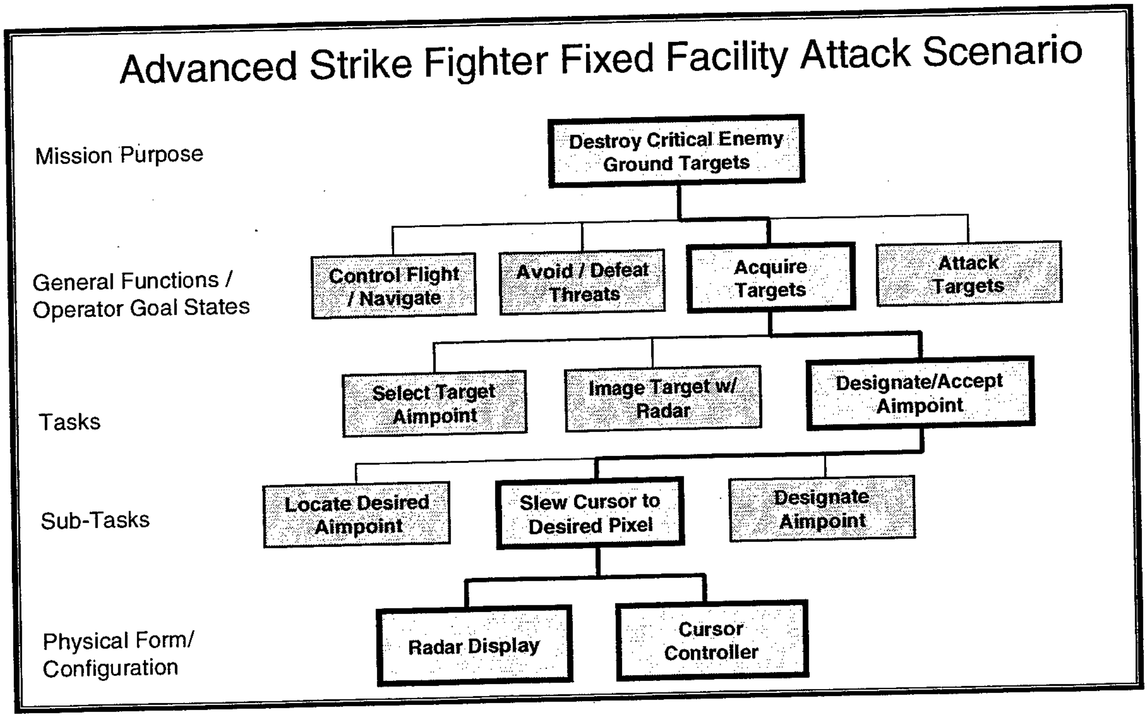


Figure 10. Excerpt from a Means-Ends Hierarchy

Specification of Goal State Definition and Interaction. Since the CART approach incorporates a goal-directed representation of human behavior, the next step in CART implementation involves further examining the operator goal states. As depicted in Figure 11, each goal state identified in the mission decomposition is given a priority and a brief description and the events that trigger its onset and offset are specified. Within the human performance model, filters will monitor for these events and trigger the appropriate operator goal states when they occur. Next, a goal state interaction matrix is developed. The goal state interaction table specifies impact of triggering a goal on all lower priority goals. The impacts are imposed only if the lower priority goal is currently active. Within the human performance model, triggering a goal state can result in other active lower priority goal states being continued, aborted, or suspended. In our hypothetical strike fighter example, activation of the *Evade Threats* goal state results in all other goal states being suspended or aborted. When the *Acquire Targets* goal state triggers, the *Attack Target* goal state will be suspended (though it is unlikely that it would be active because target acquisition should be completed before attack commences) and the *Navigate* goal state will be continued (the assumption is that navigation at this point is in support of target acquisition activities, i.e., flying a target acquisition route). Under the goal state *Target Attack*, *Navigation* is aborted because the pilot flies the airplane to the target using data from the sensors and tactical situation display.

Goal State Definition

Operator Goal States	**Priority**	**Description**	**Onset Event**	**Offset Event**
Evade Threats	1	Maneuver A/C and/or Release Chaff/Flares	Threat First Indicated on Planned Route	Threat No Longer Indicated on Planned Route
Acquire Target	2	Image, Detect, ID & Designate Target	Enter Sensor Range	Accept Aimpoint
Attack Target	3	Select, Arm, & Release Weapon(s)	Weapon Select	Weapon Release
Control Flight/Nav	4	Keep/Maneuver Aircraft to Planned Route	Continuous	Continuous

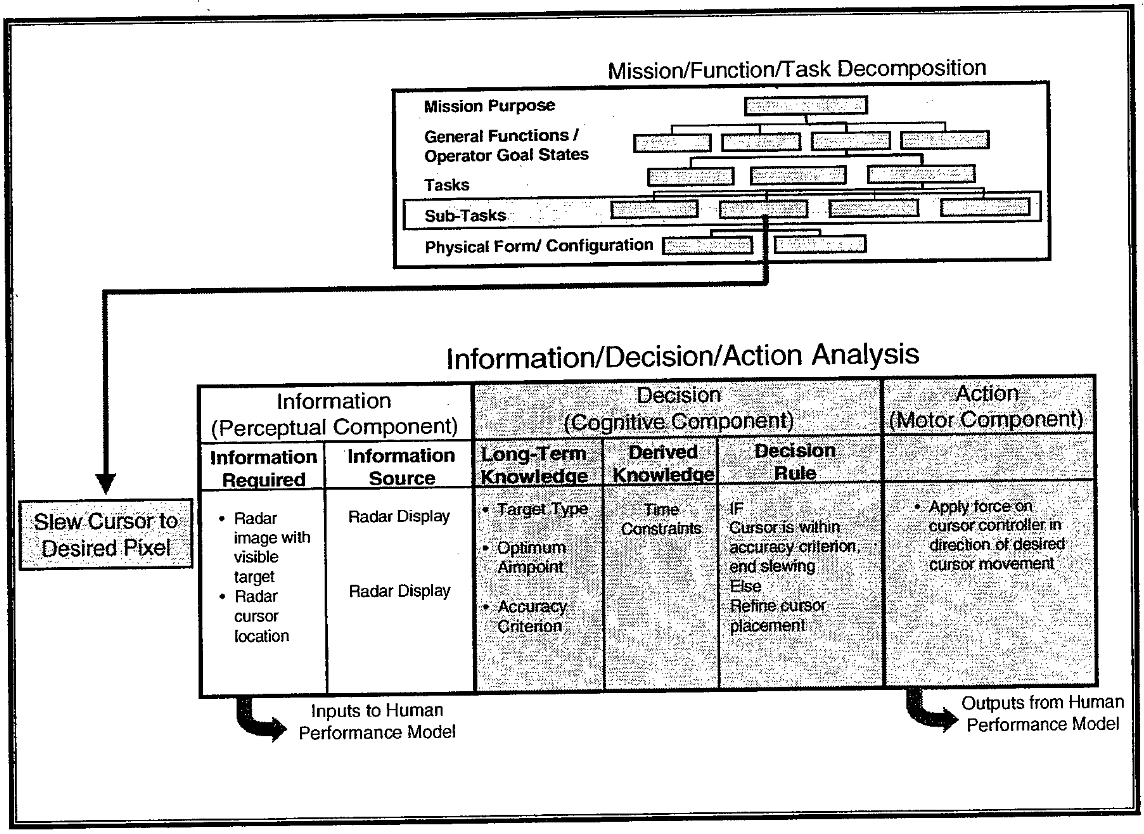
Goal State Interaction Matrix

Triggered Goal State	**If Evade Threats Active**	**If Acquire Tgt Active**	**If Attack Tgt Active**	**If Control Flight/Nav Active**
Evade Threats	▨	Suspend	Suspend	Abort
Acquire Target	▨	▨	Suspend	Continue
Attack Target	▨	▨	▨	Abort
Control Flight/Nav	▨	▨	▨	▨

Figure 11. Goal State Definitions and Interactions

Information-Decision-Action Analysis. Another key decomposition activity required to implement the CART concept is an information-decision-action analysis. Following, or in conjunction with, the mission decomposition, essential information elements, decision logic, and actions must be identified for each subtask. These relate to the human perceptual, cognitive, and motor activities, respectively, performed by the operator. This information will help model developers identify the required inputs to and outputs from the human performance model, as well as the decision rules to be implemented.

Figure 12 shows an example of information-decision-action analysis results for the *Slew Cursor to Desired Pixel* subtask listed as a sub-task in Figure 10. In this analysis, the perceptual information consists of a radar image that includes the target and a cursor location on the image. It also identifies any knowledge the operator must have in order to adequately perform the task, such as the optimum aimpoint and a criterion for designation accuracy. To support development of decision algorithms within the human performance model, this analysis also specifies the decision logic used in carrying out the task. Next, the action component of the analysis identifies the physical pilot actions that occur during task performance. These actions, which will be represented as outputs from the human performance models, will be made available to relevant subsystem models as operator control inputs.



Information (Perceptual Component)		Decision (Cognitive Component)			Action (Motor Component)
Information Required	**Information Source**	**Long-Term Knowledge**	**Derived Knowledge**	**Decision Rule**	
• Radar image with visible target	Radar Display	• Target Type • Optimum Aimpoint • Accuracy Criterion	Time Constraints	IF Cursor is within accuracy criterion, end slewing Else Refine cursor placement	• Apply force on cursor controller in direction of desired cursor movement
• Radar cursor location	Radar Display				

Figure 12. Information-Decision-Action Analysis

<u>*MOE Extension and Data Collection Definition*</u>. The last of the decomposition activities involves expanding the MOEs identified by the acquisition program and also developing detailed data collection requirements for the simulation trials (see Figure 13). The MOEs initially provided by the acquisition program will likely be at a high level, addressing measures of effectiveness to achieve mission success. In the strike fighter example, these may include such measures as the percentage of targets destroyed and the percentage of aircraft destroyed or damaged. However, to fully understand how particular task and subtask performances interact to determine these mission effectiveness measures, more detailed measures must be developed.

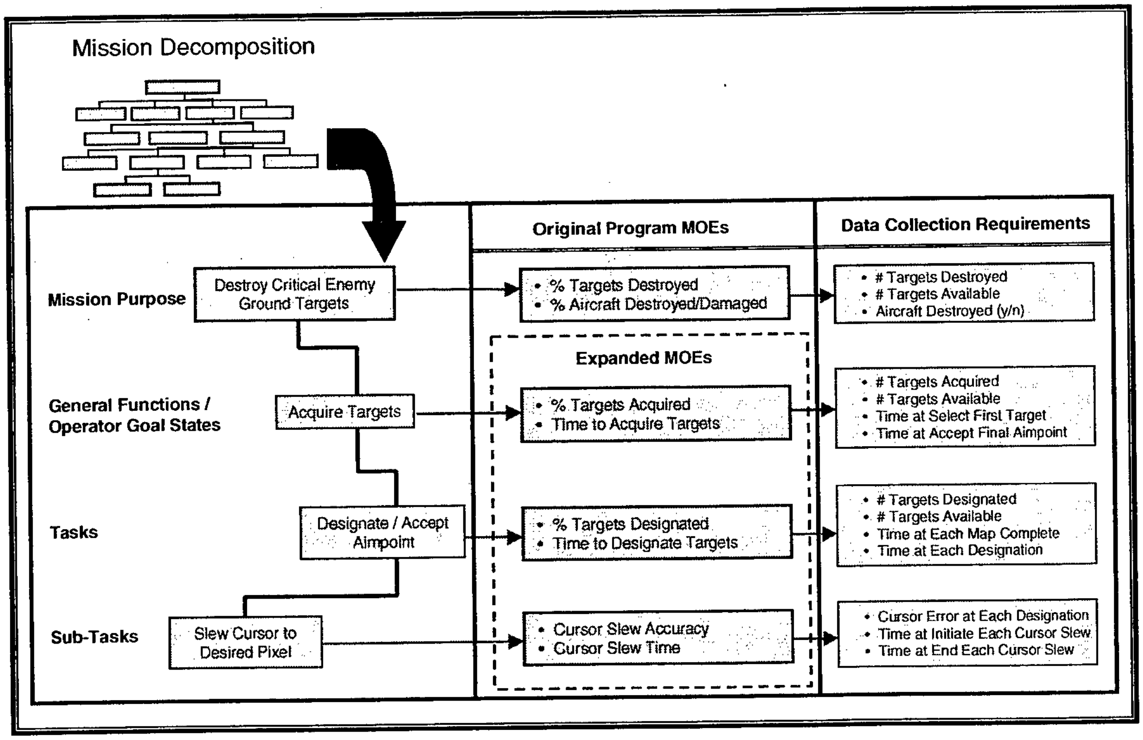


Figure 13. Example of MOE Extension and Definition of Data Collection Requirements

The MOE extension activity consists of developing a specific metric or set of metrics that quantify performance for each function, task and subtask identified in the means-ends decomposition. In specifying metrics, the objective is to provide an assessment of the critical attributes or dimensions of performance of a function, task, or subtask expected to influence mission success or failure. The resulting MOE hierarchical structure maps directly to the hierarchical components of the means-ends decomposition, and ensures that the high-level MOEs that the acquisition program has deemed most important can be traced directly to MOEs of the lowest level task performance (sometimes called Measures of Performance or MOPs). In Figure 13, MOE extension begins at the generalized function level. The generalized function *Target Acquisition* is measured by the percentage of targets acquired and the time to acquire the targets. At the task level, the task *Designate / Accept Aimpoint* is measured in terms of percent targets designated and time to complete designation. Finally, at the subtask level, the metrics center on quantifying performance of *Slew Cursor to Desired Pixel* within the designation task.

The last step in MOE specification consists of specifying all of the data collection requirements for the simulation runs. These requirements are simply the explicit data elements required to compute the extended MOEs. A collection strategy must be specified for each element (e.g., should data sampling be time-based or event-based?), sources for the data within simulation models must be identified, and a structure for storing the data must be defined. If all data elements are not readily available within the simulation, requirements will need to be specified for providing the necessary data collection software. Actual software development will occur during testbed development and extension activities described in the next section.

5.2.2 Human Performance Model Development and Integration

Once key functions, tasks, subtasks and equipment essential to mission performance have been identified by the mission decomposition activities, development of human performance models and integration of those models with the constructive system and mission environment testbed can begin. Each of these activities is described below.

Human Performance Model Development. As described earlier, the CART concept employs a task network modeling approach to human performance model development, shown in Figure 14. Using a modified version of the IMPRINT task network-modeling environment, users begin to develop a series of task networks representing the functions, tasks and subtasks identified in the mission decomposition. The execution and sequence of these networks can be either deterministic or probabilistic depending upon the type of activity modeled. Associated with each node of the task network is a set of user-defined parameters in which a number of performance values and ranges can be specified -- including the task time, accuracy, and ratings of operator workload associated with the task. The workload ratings are multidimensional, allowing the user to assign ratings independently across the visual, auditory, cognitive and physical dimensions of workload.

In developing the human performance component of the task network models, users can draw upon a number of sources to assign values to the human performance and workload variables. First, they can draw upon micro-models resident within IMPRINT. These first-principle models of human performance contain empirically-based measures of human performance for several basic perceptual, cognitive, and motor activities. For more complex or system-specific tasks, users can rely upon empirical data in the human performance literature that address the specific task of interest (e.g., a technical report that documents target designation time and accuracy associated with a thumb-actuated isometric cursor control mechanism). Where simulation resources are available, performance data can be obtained from human-in-the-loop simulation. Finally, when no such data are available, users must rely upon performance estimates from operational subject matter experts (e.g., experienced strike fighter pilots who provide time, error rate, and workload estimates for a weapon re-assignment task.)

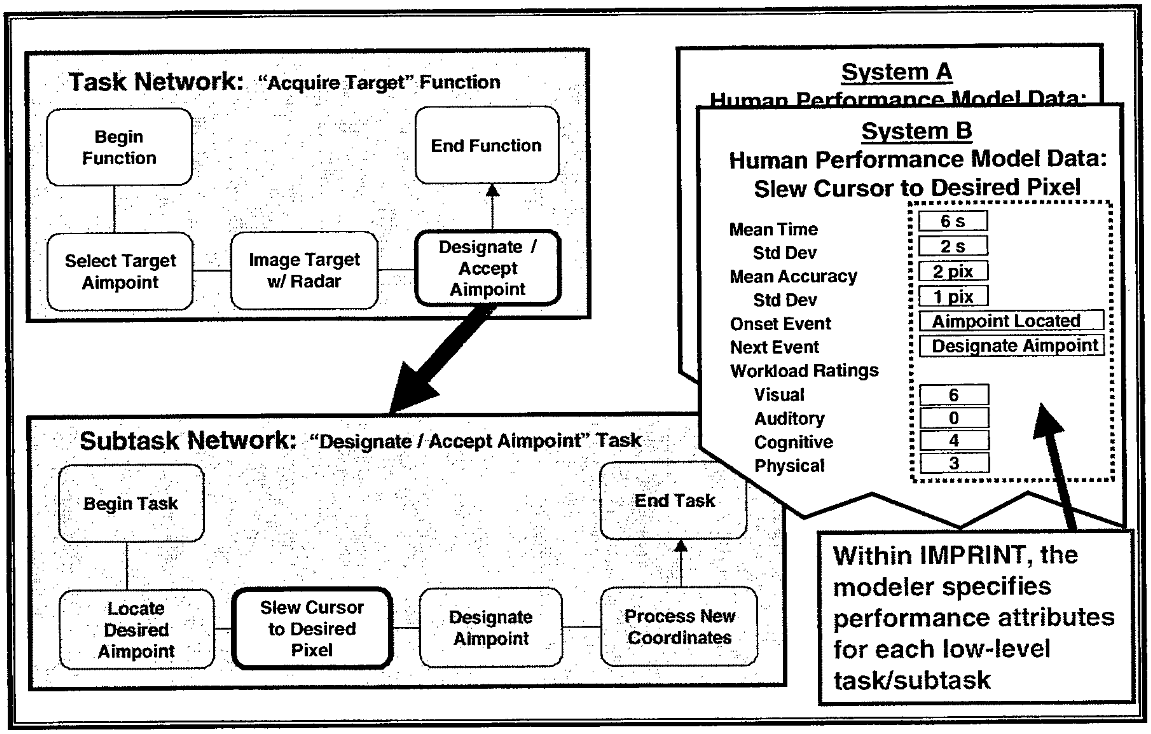


Figure 14. Human Performance Model Development

Returning to the strike fighter example, Figure 14 shows a simple task network for the *Acquire Targets* function and a subtask network for the *Designate / Accept Aimpoint* task. Once the *Designate / Accept Aimpoint* node on the task network is activated, its subtask network begins to execute. As the model runs, performance on the *Slew Cursor to Desired Aimpoint* task will reflect the range of performance (speed and accuracy) and workload values specified by the modeler.

It should be reiterated that human performance models generated using the CART approach will not be general models of human behavior applicable to all scenarios and tasks. Rather, they will be developed to represent human actions and decisions within the specific scenario(s) defined by the acquisition program. This scenario-specific modeling will allow a maximum level of model detail, resulting in a greater degree of model fidelity for a relatively low level of effort to develop the model. Even though CART human performance models are expected to be narrowly-focused to support specific test scenarios, the ease of use of the IMPRINT interface and the relative simplicity of task network modeling mean the CART models developed for one purpose can be adapted readily and extended to support new or changing study requirements. This means that an initial investment in a model can be leveraged extensively as the model is modified and reused.

Human Performance Model Integration. Once the human performance models have been completed, efforts will focus on integrating the models to run within the simulation program's constructive simulation environment. The CART concept assumes that the program is using the HLA framework, which allows a number of individual simulations to interact within a single federation. To develop the federation, there is a five-step process (*High Level Architecture*, 1998), which includes determining federation requirements, developing conceptual models, designing and developing the federation, integrating and testing the federation, and executing the simulation (DOD 5000.59-P, 1995). As shown in Figure 15, the majority of the CART integration effort will center on design and development of the federation.

As discussed earlier, key components of the HLA approach include SOMs and FOMs. A SOM exists for each simulation in the federation and specifies the information types that the particular simulation will make available to other simulations in the federation, whereas the FOM can be thought of as a 'master list' that specifies all shared information within a federation. Integration activities will focus on tailoring the FOM and SOMs to accommodate data inputs and outputs required by the human performance model. As part of the information-decision-action analysis conducted during the task decomposition activities, both the required inputs to and outputs from the human performance model will have been defined. The inputs consist of *external* events such as threat radar mode changes or missile launches by the air defense site that must be made available to the perceptual network of the human performance model. The outputs represent results of human performance model algorithms that trigger an operator action that alters the system state. For example, a pilot action to release chaff, generated in the human performance model, would result in a message to the aircraft system model to initiate a chaff release event.

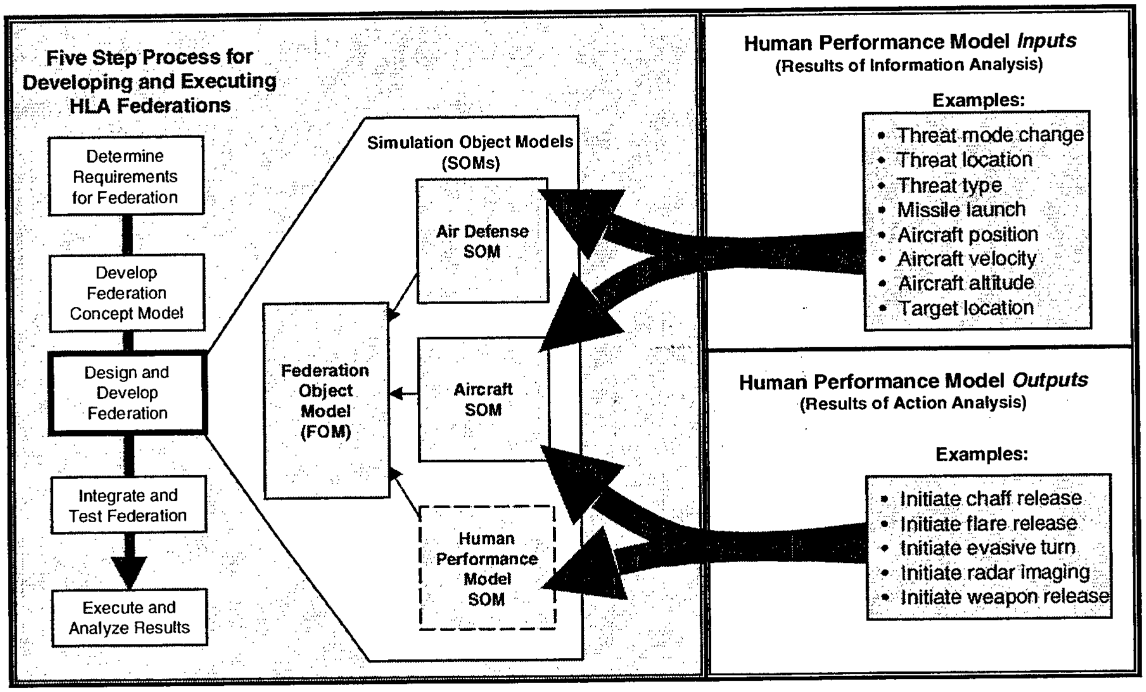


Figure 15. Integrating HPM with the HLA Federation

Key integration activities will include ensuring that existing SOMs provide the input data required by the human performance model, developing a new SOM for the human performance model itself, and subsequently modifying the FOM to incorporate all data from the modified SOMs. In the event that the human performance model requires input data that are not available from any other models in the federation, a decision must be made whether to modify one or more of the other models to provide the required data or to modify the human performance model to eliminate the data requirement.

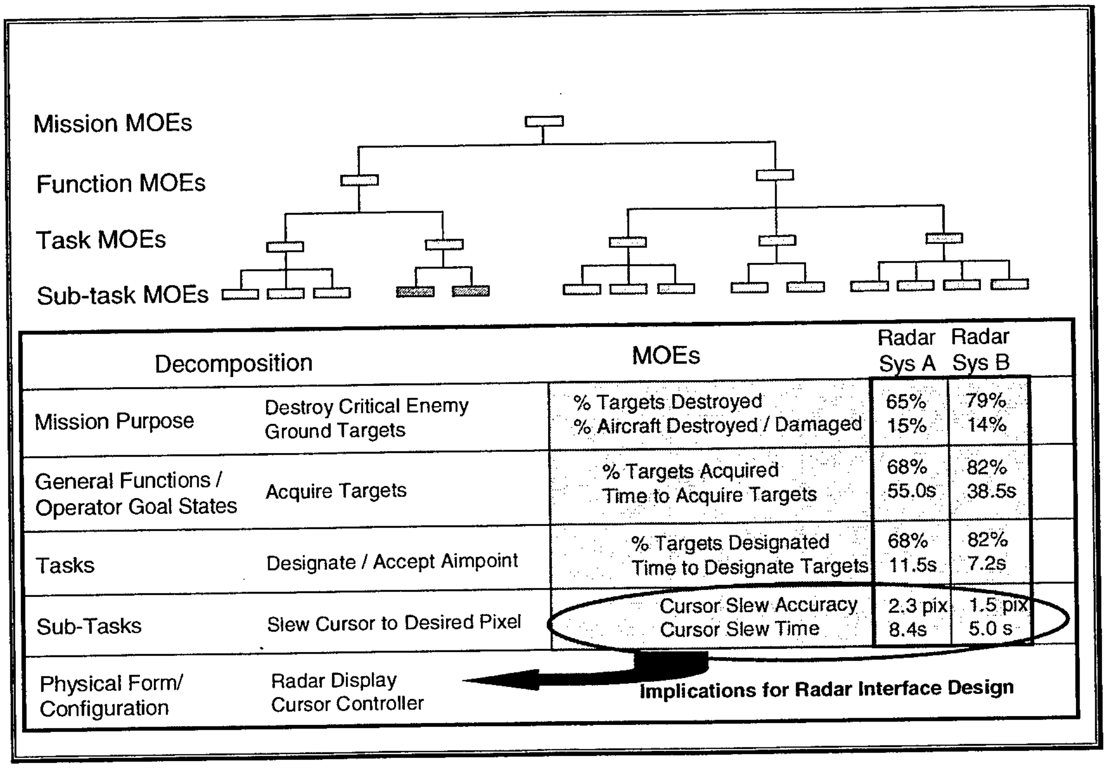
5.2.3 Constructive Simulation and Data Analysis

Upon completion of the model integration activities and subsequent testing, the federation will be ready to support the acquisition program's simulation activities. In preparation for testing, a matrix is developed that outlines all of the different test cases that are to be examined in the simulation trials, specifying the levels of each variable to be manipulated. Multiple simulation runs are then performed within each particular test case. The savings in time and cost that can be realized by using constructive human representations instead of live subjects is one significant advantage that this approach offers. Further, because trials can be run faster and easier with constructive models -- with greater variability of initial conditions -- greater numbers of trials can be run within each test condition, leading to greater statistical validity of simulation results.

As the simulation trials are run, data collection routines collect all of the required performance measures necessary to calculate the expanded MOE / MOPs described earlier. Explicit and derived MOE data are then compiled for each test condition and formatted for subsequent data analysis.

CART's implementation of a human performance representation within the constructive simulation environment will not change the high-level goals and approach of the traditional data analysis used in ultimately generating requirements. That is, the analysis will still focus on identifying differences in mission outcomes (reflected in mission-level MOEs) achieved by the various system concepts being addressed. The CART approach will simply allow a more detailed data analysis, providing traceability of mission results to particular task performance and crew system components. If differences in mission-level outcomes are identified, the analysis then turns to the lower levels of the MOE hierarchy to identify which function, task, and subtask performance measures best explain the differences in mission outcome.

For example, the hypothetical analysis shown in Figure 16 found that the use of Radar System B in the strike fighter resulted in a greater degree of mission success than Radar System A. Looking at the function level, it was determined that varying the radar system had the greatest impact on the target acquisition function. Turning to the tasks that support the target acquisition function, aimpoint designation performance was found to vary substantially between the two radar systems. Finally, the best explanation for designation performance differences was found to be the time and accuracy associated with cursor slewing. With this insight of how low-level task performance can significantly impact overall mission outcomes, the analyst will be able to suggest focused, performance-based requirements for the crew system interfaces used in performing the key tasks and subtasks.



Decomposition		MOEs	Radar Sys A	Radar Sys B
Mission Purpose	Destroy Critical Enemy Ground Targets	% Targets Destroyed % Aircraft Destroyed / Damaged	65% 15%	79% 14%
General Functions / Operator Goal States	Acquire Targets	% Targets Acquired Time to Acquire Targets	68% 55.0s	82% 38.5s
Tasks	Designate / Accept Aimpoint	% Targets Designated Time to Designate Targets	68% 11.5s	82% 7.2s
Sub-Tasks	Slew Cursor to Desired Pixel	Cursor Slew Accuracy Cursor Slew Time	2.3 pix 8.4s	1.5 pix 5.0 s
Physical Form/ Configuration	Radar Display Cursor Controller	Implications for Radar Interface Design		

Figure 16. Data Analysis

5.2.4 CART-Based Crew System Performance Requirements

Based on the identified linkage between cursor slewing performance during target designation and overall mission outcomes shown in Figure 16, a performance-based requirement for aimpoint designation can be specified. For example:

Requirement 1. *The crew system interface for the targeting radar system shall support accurate and rapid slewing of the radar cursor.*

a. *The system shall support target designation accurate to within 1.5 pixels.*

b. *The system shall support target designation times not to exceed 5.0 seconds / target.*

A requirement such as this specifies a given designation accuracy and time that the crew system must support in order to achieve the desired level of mission performance, without specifying any particular physical design or detailed characteristics of the system. Such a requirement, developed relatively early in the acquisition program, would provide a clear metric against which crew system designers could later evaluate various design options. This would also serve to limit the search space so that subsequent human-in-the-loop (HITL) simulations would be able to focus on the most promising crew system concepts.

6.0 CART COSTS AND BENEFITS

Assessment of the utility of a capability such as CART requires the joint consideration of the costs of using the capability and the benefits that accrue from such use. An objective, quantitative cost and benefit assessment of CART is difficult because costs depend on how a user expects to apply CART and data on benefits are very difficult to obtain. Nevertheless, a discussion can be provided that describes the factors that drive CART cost and the benefits that accrue from using CART. Hopefully, this discussion will be sufficient to convey that, for a relatively small investment, CART improves the crew system design process resulting in significant cost and time savings associated with remediating crew system design flaws, as well as, providing additional benefits described below.

6.1 CART Costs

As indicated above, implementation and application costs of CART will vary, depending on the specific needs and circumstances of the user. Costs, however, can be grouped into four factors: human performance model development, HLA integration, constructive simulation extension, and test execution and analysis. Each factor is discussed below.

6.1.1 Human Performance Model Development

The development costs of the human performance model consist of the time and labor to conduct the mission decomposition activities described earlier, and the creation of the task network models that are the HPM. Creation of the HPM includes integration and test activities with the constructive simulation environment. If first-principle models are required, their development or modification and integration into HLA would also have an associated cost. While mission and task complexity is the major cost driver here, model development is generally a non-recurring cost that has the benefit of extensive reuse (with minor modifications) across multiple test events. The graphical nature of the IMPRINT user interface makes HPM development a relatively easy process, reducing labor required. Another possible way to reduce cost is to reuse a model developed by another user. In this case, costs would be limited to the effort required to modify the model. The open, HLA interface used by CART makes sharing and reuse of models across different testbed environments a relatively easy process.

6.1.2 HLA Integration

CART assumes that the user has an existing constructive simulation environment. Integrating CART requires creating an HLA interface between CART and the constructive simulation. If the constructive simulation is already HLA compliant, that capability will need to be extended to provide the data needed by the human performance model. Costs here fall into the two categories of analysis and software development. Analysis consists of the FOM and SOM development activities described in the previous section. This process results in a complete understanding of the data that needs to be exchanged between the HPM and the simulation environment. Software development involves modifying constructive simulation environment models to support the data exchange requirements dictated by the FOM and SOM. The magnitude of this effort depends on factors such as the amount of data that must be manipulated and the difficulty of getting data into and out of the constructive simulation environment. HLA integration is generally a one-time cost, however user-developed SOMs and FOMs are re-useable and interoperable with future simulation exercises.

6.1.3 Constructive Simulation Augmentation

In some instances, it may be necessary to add capabilities to the existing constructive simulation environment to meet particular requirements of CART. One of these areas is data collection. If CART MOE / MOPs require data from the constructive simulation environment that are not routinely collected, it will be necessary to extend the data collection functions to include these data. Another potential area of augmentation is enabling the constructive system model (e.g., an aircraft model) to implement actions directed by the CART HPM. For example, if the CART HPM is to 'fly' an aircraft, the constructive system model might need to be modified to implement control operations from the CART HPM (e.g., replace stick and rudder movements with a macro capability that implements the maneuvers at a higher level -- such as 'turn right 30 degrees' or 'descend 1000 feet'). Constructive simulation extension is also a one-time cost.

6.1.4 Test Execution and Analysis

Recurring costs of CART use are found in the application of the fully-integrated CART to conduct trade studies. CART requires a small, incremental cost over and above the baseline costs associated with employing the constructive system simulation. These costs are driven primarily by the additional data reduction and analysis activities required to compute and analyze CART MOE / MOPs.

6.2 CART Benefits

The discussion of CART benefits is divided into two subsections. The first section discusses benefits that accrue to the system acquisition process in general. The second section is more focused on how CART will change the use of HITL simulation in the acquisition process.

6.2.1 CART Benefits for System Acquisition

The ability to insert human performance models into constructive simulation environments now means that the *complete* system (i.e. one including the operator) can be represented using constructive models. Explicit consideration of the crew system can be incorporated into the trade studies conducted early in the acquisition process and crew system performance requirements can be specified along with other system requirements. Consequently, the data derived from the system simulation as a whole should be more accurate and lead to specification of more achievable operational requirements for all components of the system.

The real benefit of CART extends beyond specification of crew system requirements. It will be the quality of the crew system designs that result from those requirements. Having requirements that specify levels of performance that must be achieved by an interface will focus crew system design activities. They will provide benchmarks against which designs can be tested, making it easier to discover problems in the PDRR and EMD phases rather than later during operational test and fielding. As such, implementation of CART will result in more effective crew system designs with fewer design flaws that require subsequent redesign efforts. This, in turn, will save time and money in the acquisition process.

In addition to facilitating development of more effective crew systems, CART will aid other aspects of system fielding. Development and application of human performance models should provide system developers with an intimate understanding of the skills and abilities required in a job, mission critical performances, and tasks that have significant workload and situation awareness impacts. This information should aid in the development of more effective operator training as CART's performance measurement structures, which link operator performance to mission outcomes, also has a training application. CART's derived MOPs and MOEs can be used to evaluate human operators and provide feedback that helps them better understand how their performance impacts mission outcomes.

6.2.2 CART's Impact on the Use of Human-in-the-Loop Simulation

CART will change how human-in-the-loop simulation is used in the acquisition process. In the *absence* of effective human performance modeling, many acquisition programs have resorted to extensive use of human-in-the-loop simulation to assess operator function and task allocation, to develop component-oriented (as opposed to performance-based) requirements, and to explore crew system design issues. HITL simulation has intuitive appeal because the data generated by the simulation seems to be inherently valid (it is generated by humans, ergo, it is a valid representation of human performance). HITL simulation does, however, have its limitations.

HITL simulation is expensive and time consuming to develop, employ, operate and maintain. A particularly costly HITL component is development of the operator interface (controls and displays). This can be an ongoing cost as the interface is changed repeatedly to reflect different system alternatives. With CART, human performance models (which do not need an expensive functional crew system interface) can be used to assess the complete range of system alternatives proposed. HITL simulation can be reserved for testing the subset of alternatives deemed most promising. This will reduce the number of alternatives that need to be represented in HITL testing, thus reducing HITL simulation development costs. Also, the need for numerous personnel to support data collection trials (e.g., subjects, controllers, and actors to play opposing force and command and control roles) -- coupled with the time consumed by those trials -- is a significant cost driver for HITL simulation employment. Constructive simulations, on the other hand, can be run in a batch mode with very little human supervision. Finally, HITL simulations must run in real-time; this becomes a limiting factor when compared to constructive simulations that can run much faster than real-time, generating many trials in the time it takes HITL to generate one trial.

Other issues surround the validity of the data obtained using HITL simulation. As force sizes have been reduced, it has become increasingly difficult to obtain subjects for HITL tests. As a result, numbers of subjects used in HITL simulation are often too small for inferential statistical testing, or the subjects used do not represent the target pilot population. Another problem is subject proficiency. HITL testing often occurs over a relatively short time span (days or weeks). For systems that the subjects have not previously seen, significant training time is required to develop proficiency in system operation and employment. Given the limited exposure to the new system, subjects often do not have the time to become proficient. In these instances it is difficult to determine how representative HITL subject performance is of the proficient personnel who ultimately will employ the system. When developing crew system requirements, the HITL simulator's operator interface may itself confound the results. While an operator interface is a necessary component of HITL simulation, it must be recognized that the interface will have an impact on the observed operator and mission performance. The performance observed may not properly reflect the system's overall performance because the interface used in the simulation is not the interface that will ultimately be in the system fielded. The CART concept seeks to mitigate this potential problem by using human performance modeling to determine levels of crew system performance required to achieve desired mission outcomes, and then drive the crew system design process to produce interfaces that yield the user's required performance levels.

Finally, it is not expected that CART will replace the use of HITL simulation. Despite the limitations described above, HITL simulation is an important element of the acquisition process. Giving operators hands-on experience is critical for gaining operator feedback about the new system (e.g., options for tactics, or the usability of designs and features). With CART, HITL simulation offers a means of verifying and validating the results of CART task allocation and crew system requirements development. A goal of CART is to reduce the total amount of HITL simulation required in system acquisition, and to provide a better focus for those HITL activities that are later performed as subsystem components are designed or selected.

7.0 SUMMARY AND CONCLUSIONS

CART provides a human performance modeling approach to generating performance-based crew system requirements. CART will lead to more effective crew system designs that reduce the cost and effort associated with correcting flaws in the weapon system once fielded. As an innovative application of human performance modeling, CART advances the objective of providing an authoritative representation of human behavior established in the DOD Modeling and Simulation Master Plan (DOD 5000.59-P, 1995) published by DMSO. CART's use of the HLA as its interface to other simulations and the ready reusability of its human performance models advances Simulation Based Acquisition (SBA) objectives of collaboration and reuse, as well as its broader objective of providing better, cheaper, and faster acquisition.

In a major review of the current state of human behavioral representation, Pew and Mavor (1998) note that, in general, "user expectations exceed Human Behavioral Representation (HBR) capabilities" ... and ... "HBR falls short of user needs." While we believe this is true for many applications of human performance modeling, we think that the generation of crew system requirements is an exception. Requirements generation occurs as part of the system concept definition and refinement process through analyses and trade-offs; during this time, system capabilities are generally specified in terms of functional requirements rather than candidate design configurations. The human performances contemplated in the system can be represented at relatively low resolution because there is often little detail as to how these system functions will actually be implemented. Also, since requirements generation focuses upon identifying and manipulating important attributes and levels of performance, the impetus to simulate underlying cognitive and behavioral processes is relaxed. Taken together, these factors mean that relatively simple representations of human performance can be applied successfully to generate performance-based crew system requirements. Thus, CART's focused application of human performance modeling offers the opportunity for a demonstration of the technology that truly adds value to the warfighter.

8.0 BIBLIOGRAPHY

Air Force Instruction 10-601 (1 October 1998). *Mission Needs and Operational Requirements Guidance and Procedures.*

Air Force Modeling and Simulation: A New Vector (June 1995). Report from the Air Force Four-Star M&S Summit.

Aldrich, T.B., W. Craddock, and J. H. McCracken (1984). *A Computer Analysis To Predict Crew Workload During LHX Scout-Attack Missions: Volume 1* (MDA903-81-C-0504/ASI479-054-I-84(B)). United States Army Research Institute Field Unit, Fort Rucker AL.

DOD 5000.59-P (October 1995). *Modeling and Simulation Master Plan.* Defense Modeling & Simulation Office, Alexandria VA 22311.

Doyal, A. J., G. E. Irvin, and D. P. Ramer (1995). *Operator Cursor Positioning Performance On Navigational Update and Targeting Tasks Evaluation of Gain Functions for the B-2 Radar-Embedded Cursor System* (US Air Force Armstrong Laboratory Technical Report Number AL/CF-TR-1995-0106). Wright Patterson Air Force Base, OH 45433-7022.

Farmer, E. W. , A. J. Belyavin, S. G. Russell, D. M. Jones, and C. L. Birch (1995). *Prediction of Operator Workload and Performance in Future Systems* (Defence Evaluation and Research Agency Draft Technical Report). Farnborough Hampshire, United Kingdom.

Flach, J. M. (1990). Control with an eye for perception: Precursors to an active psychophysics. *Ecological Psychology.* 2, 83 – 111.

Flach, J. M. and C. O. Dominguez (1995). Use-Centered Design: Integrating the User, Instrument, and Goal. *Ergonomics in Design, July 1995,* 19-24.

Frost, R. and D. Thomen (1998). Simulation Based Acquisition. *Proceedings of the Spring 1998 Simulation Interoperability Workshop,* Paper Number 98S-SIW-228.

Hendy, K. C. and P. S. E. Farrell (1997). *Implementing a Model of Human Information Processing in a Task Network Simulation Environment* (Canadian Department of National Defence, Defence and Civil Institute of Environmental Medicine Technical Report DCIEM No. 97-R-71). North York, Ontario, Canada M3M 3B9.

High Level Architecture [A Primer on CD-ROM] (March 1998). Defense Modeling & Simulation Office, Alexandria VA 22311.

Improved Performance Research Integration Tool (IMPRINT) User's Guide Version 4.0 (April 1998). US Army Research Laboratory, Human Research and Engineering Directorate, Aberdeen Proving Ground MD 21005-5425.

Pew, R. W. and A. S. Mavor (Eds.) (1998). *Modeling Human and Organizational Behavior.* Washington, D.C.: National Academy Press.

Pope, M. F. (1999). A CGF to meet the existing and future training requirements of the B-2 Aircrew Training Device. *Proceedings of the Eight Conference on Computer Generated Forces and Behavioral representation,* Paper Number 8^{TH}-CGF-093.

Rasmussen, J., A. M. Pejtersen, and L. P. Goodstein (1994). *Cognitive Systems Engineering.* New York: Wiley

9.0 GLOSSARY OF ACRONYMS

AOA	Analysis of Alternatives
C2	Command and Control
CART	Combat Automation Requirements Testbed
COGNET	Cognition as a Network of Tasks
CTA	Cognitive Task Analysis
DOD	Department of Defense
DMSO	Defense Modeling and Simulation Office
EMD	Engineering and Manufacturing Development
FOM	Federation Object Model
GPS	Global Positioning System
HBR	Human Behavioral Representation
HIP	Human Information Processor
HITL	Human-in-the-Loop
HLA	High Level Architecture
HPM	Human Performance Model
IMPRINT	Improved Performance Research Integration Tool
INS	Inertial Navigation System
JDAM	Joint Direct Attack Munition
MOE	Measure of Effectiveness
MOP	Measure of Performance
M&S	Modeling and Simulation
OMAR	Operator Model Architecture
OPFOR	Opposing Force
ORD	Operational Requirements Document
PDRR	Program Definition and Risk Reduction
RCM	Requirements Correlation Matrix
RCS	Radar Cross Section
RTI	Run Time Infrastructure
RWR	Radar Warning Receiver

SAM	Surface-to-Air Missile
SBA	Simulation-Based Acquisition
SOM	Simulation Object Model
SPO	System Program Office
SWEG	Simulated Warfare Environment Generator
VACP	Visual, Auditory, Cognitive, and Psychomotor
WSO	Weapon Systems Officer

APPENDIX A

A CASE FOR CART: AN ILLUSTRATION OF THE POTENTIAL BENEFITS OF USING CART IN ACQUISITION

This page left blank intentionally.

INTRODUCTION

This appendix illustrates some challenges that many current system acquisitions face, and how CART can help overcome those challenges. It will accomplish this goal by presenting an example of a subsystem acquisition program incorporated into an aircraft upgrade program. This example illustrates how an inadequate understanding of crew system interface requirements and the lack of crew station performance requirements that are directly tied to mission requirements can ultimately lead to an interface design that does not support the mission goals. This is but one example of a problem that is pervasive in weapon system acquisition. Redesign efforts generally impact cost and schedule for the program as a whole, and may be preventable once CART methods and technology are available.

Types of System Acquisitions

There are multiple types of system acquisition programs. Some acquisitions are for entirely new systems intended to fill an existing or projected need. A good example of this type of program is the Joint Strike Fighter (JSF) program. It is intended to fulfill the long-term need for a multi-role fighter aircraft for the 2010 and beyond timeframe.

Other acquisitions are intended to enhance capabilities of existing systems in major ways. For example, most aircraft undergo major block upgrades during the course of their service lives. An example of this type of acquisition is the F-16 Block 50 avionics upgrade. This type of upgrade can be a relatively major procurement in terms of cost and effort, although certainly nowhere near the cost and effort required to build a new system from the ground up.

Still other types of acquisitions are narrowly focused solutions to specific, minor system needs. Often, the solutions required to meet the acquisition program's needs already exist on other platforms. These are sometimes called product insertions. An example of a product insertion is the B-1B Towed Decoy System. The purpose of this acquisition was to improve the existing B-1B defensive avionics suite without resorting to wholesale changes to the defensive subsystem.

Regardless of the scope of these system acquisitions, even seemingly minor changes to existing systems may have significant implications for crew system design. A good example of this situation was demonstrated by the insertion of the Joint Direct Attack Munition (JDAM) into an existing aircraft system and the resultant mission performance degradations brought about by inadequately understood crew system performance requirements.

THE JDAM TECHNOLOGY INSERTION

The JDAM was itself a modification program. Specifically, JDAM was a program designed to enhance existing stockpiles of certain conventional weapons, (e.g., the Mk-84 general-purpose bomb, and the BLU-109 penetrating bomb) by making them precision-guided. It accomplished this goal using a modified tailkit that includes movable fins, allowing the bomb to change its trajectory and flight path after release. The tailkit also includes a Global Positioning System (GPS) receiver and its own Inertial Navigation System (INS) to provide guidance. The weapon's avionics is provided the coordinates of a ground location, and the tailkit's movable fins -- controlled by special drive laws -- guide the weapon to that location.

JDAM was originally designed to be deployed on ground attack fighters like the F-15E, the F-16, and the F/A-18. However, the USAF decided to also incorporate a JDAM weapon capability into a planned upgrade program that was underway for a large-payload capacity aircraft.

The plan was to mount eight, 2,000 pound JDAM bombs on the rotary launcher shown in Figure A-1. Since this aircraft could carry three rotary launchers, it could carry 24 weapons. Thus, if it were carrying the maximum number of JDAM weapons, it could precisely target up to 24 separate ground locations, significantly more than any of the previously mentioned fighter aircraft.

In an operational mission plan, weapon target assignments would be made by assigning targets to bombs. Each target in the mission plan would be assigned one or more weapons, and -- by association -- the launcher station from which those bombs would be released.

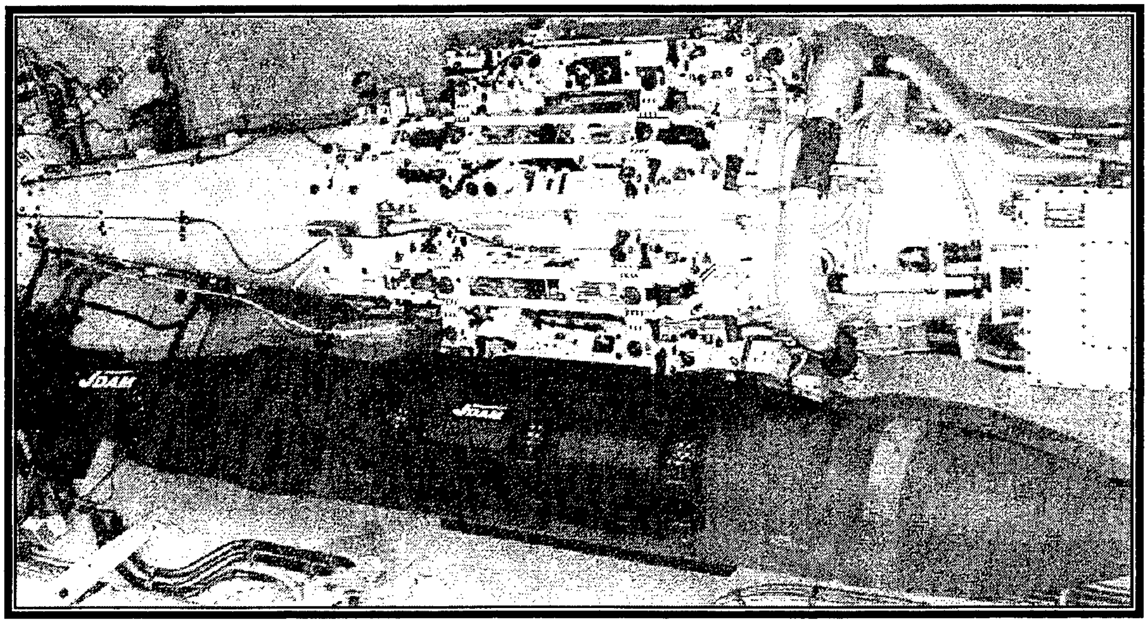


Figure A-1. A JDAM Weapon Mounted on an Eight Station Rotary Launcher.

System Characteristics and Their Operational Implications

A typical release sequence with most rotary launcher-mounted weapons would have been stations one through eight sequentially, with the opportunity to skip any or all stations. With JDAM, however, the sequence became more complicated.

Due to the size of the fins in the bomb tailkit, weapons at even-numbered stations were blocked from release by both weapons at the adjacent, odd-numbered stations. The blockage problem is illustrated in Figure A-2. The proposed workaround was to set the release order of the weapons so that all the odd-numbered weapons on a launcher were released before all the even-numbered weapons. This changed the planned weapon release order from 1-8 sequentially to 1, 3, 5, 7, 8, 6, 4, and finally 2.

If a particular mission proceeded as planned, that is, all weapons were released against their intended targets in the planned order, there was no problem. Changes in the planned sequence of weapon drops, however, necessitated changes to the weapon-target assignments.

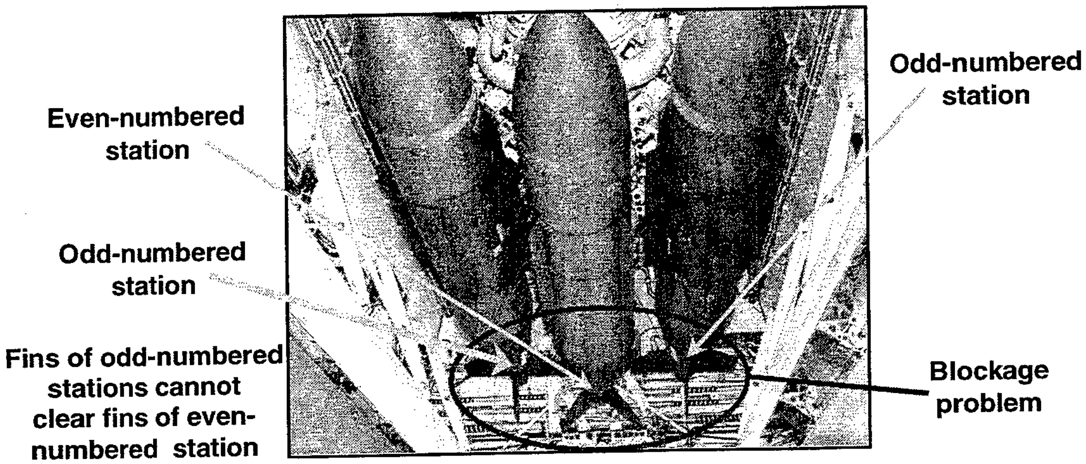
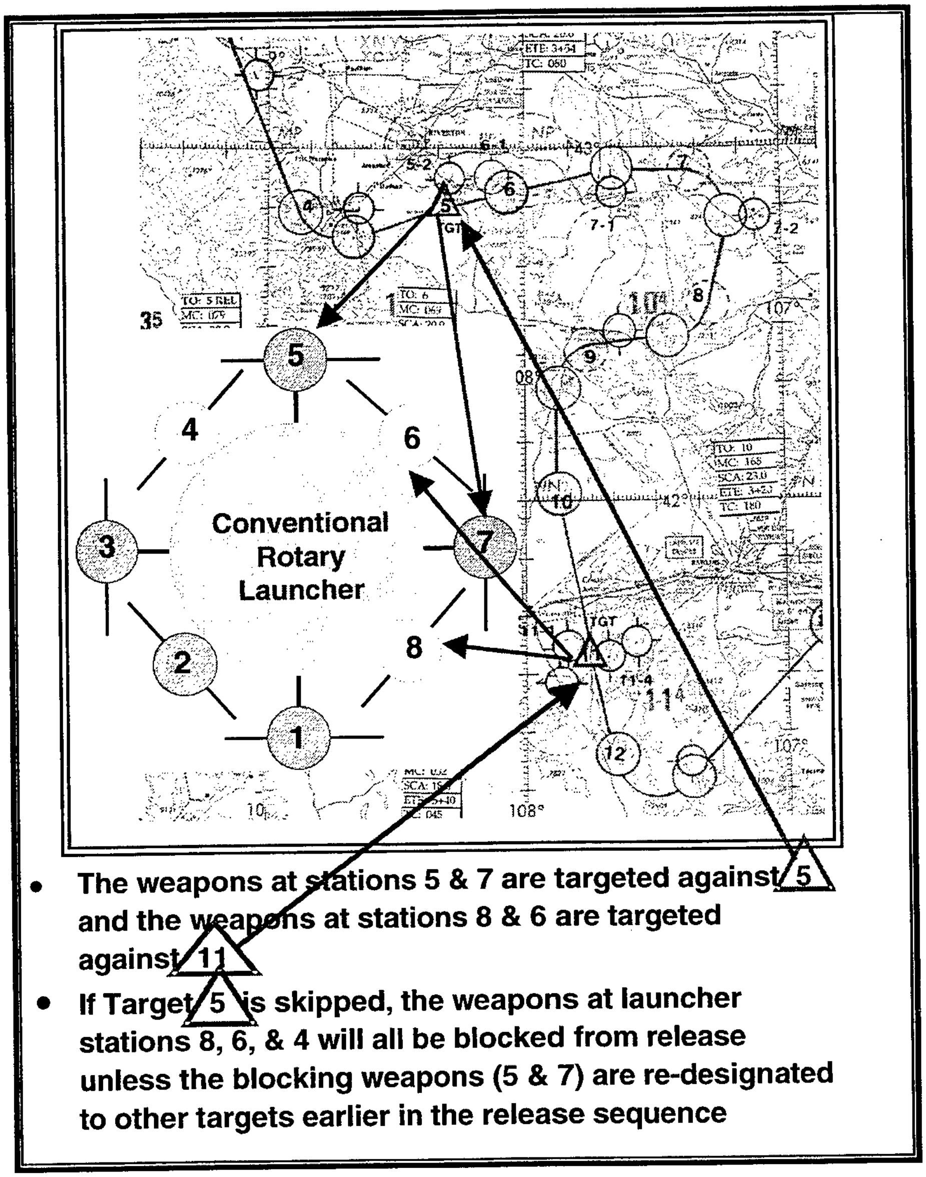


Figure A-2. A Graphical Depiction of the Blockage Problem

JDAM Employment Constraints on the Aircraft

The weapon system had a requirement for mission flexibility. That is, target sequences could be altered from the original planned sequence and target sets could be changed, all during the course of the mission. When target changes occurred, the Offensive Weapon Systems Officer (Offensive WSO) had to re-designate targets and weapons in flight. The blockage problem complicated weapon re-designation significantly for JDAM weapons. The most challenging situation involved a skipped target scenario in which a target in the originally planned mission sequence was passed or skipped without releasing a weapon against it, and the withheld weapon was then blocking the release of other weapons intended for subsequent targets (for example, see Figure A-3). In this situation, the WSO would have had to re-designate the blocking weapons by assigning them to targets appearing in the new mission sequence prior to the targets assigned to the weapons that were being blocked.

In the example illustrated in Figure A-3, the weapons mounted on stations 5 and 7 are designated for target number 5 and the weapons at stations 8 and 6 are designated for target 11. If an in-flight replan required skipping target 5, the weapons at stations 8, 6, and 4, would all be blocked from release. These weapons would remain, until the blocking weapons, 5 and 7, were redesignated against other targets earlier in the release sequence.



- **The weapons at stations 5 & 7 are targeted against 5 and the weapons at stations 8 & 6 are targeted against 11**
- **If Target 5 is skipped, the weapons at launcher stations 8, 6, & 4 will all be blocked from release unless the blocking weapons (5 & 7) are re-designated to other targets earlier in the release sequence**

Figure A-3. A Description of the Skipped Target Problem

Interface Design Constraints

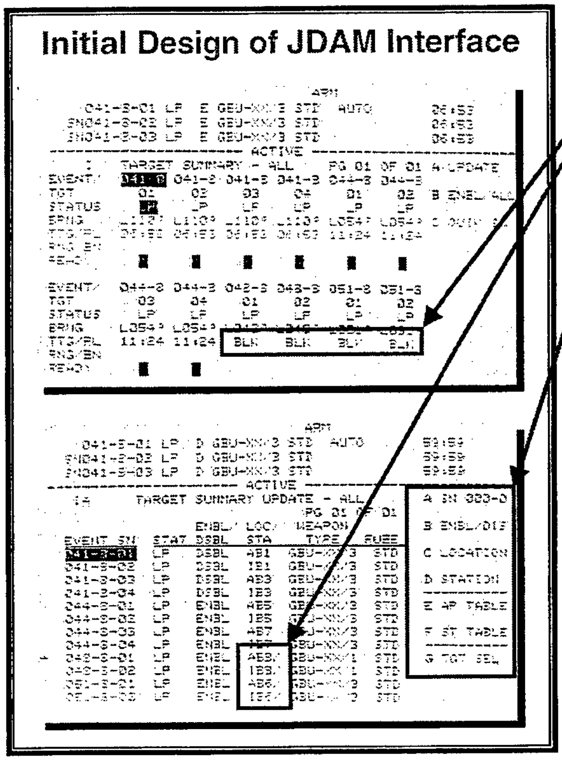
Due to cost and schedule pressures, severe constraints were levied on interface designers attempting to integrate the JDAM capability into the aircraft upgrade. These *constraints* included *no changes to the existing physical controls and displays*, the *reuse of* as much of the *existing software interface* as possible, and *minimizing the impact to other software* as part of a larger upgrade program. With these constraints, the approach to the JDAM interface design was to reuse an existing weapon assignment display format with only slight modifications and additions.

While this approach met the programmatic constraints for development, it de-emphasized consideration of the total human-system performance.

Interface designers recognized that the potential for blockages impacted weapon re-designation task performance, and that the JDAM re-designation interface needed to provide the WSO with information on which weapons were blocked. Unfortunately, the information that could be made available without a major upgrade provided only a surface understanding of the blockage problem, and did little to aid the operator.

Physical Blockage vs. Logical Blockage

As far as any mission was concerned, a weapon blockage did not really exist as long as the weapon release sequence continued to release weapons at odd-numbered stations before moving to even-numbered stations. Even though a weapon on an odd-numbered station was physically blocking the adjacent even numbered stations at a point in time, this was unimportant as long as the release sequence supported the release of the odd-numbered weapons before release of the even-numbered weapons. The important piece of information to the WSO was whether or not he had a *logical* blockage -- that is, a blockage that was being caused by a release-sequencing problem. Unfortunately, it was information about *physical* blockages that was provided in the re-designation display (see Figure A-4). With the use of the 'BLK' indication on one display and a slash next to the blocked weapon on another, this interface cued the operator that a physical weapon blockage existed. This had the potential impact of cueing the WSO that a blockage problem existed when, in fact, the sequence had no logical blockages. Information about logical blockages was not displayed and as such, the logical blockages had to be manually re-calculated by the WSO. The resulting displays failed to adequately support the decision process because they did not provide the logical blockage information critical to solving the re-designation problem.



Initial Design of JDAM Interface

- **Physical blockage indications were displayed**
- **These controls provided the Weapon Systems Officer (WSO) the capability to *implement* a re-designation decision**
- **Nothing in the interface aided the WSO in the *re-designation decision process***

Figure A-4. Initial Human Interface for JDAM

The challenge for the WSO was to be able to mentally project ahead in a weapon delivery sequence and determine whether a real blockage problem existed. If blockage did exist, the WSO needed to be able to alter the sequence and project its impact. This cycle of modifying weapons sequences and projecting impacts continued until a satisfactory sequence was obtained. *Physical blockage* information was of no use in this process, only *logical blockage* information could help.

Other Interface Considerations

In addition to the lack of logical blockage information, nothing implicit or explicit in the interface gave the WSO true insight into the correct re-designation solution. Also, nothing inherent in the interface provided the WSO with appropriate feedback that he was on the right track. The cognitive complexity of solving a logical blockage problem with this interface was soon evident and the difficulty of redesignating these weapons increased as other factors such as different bomb bodies and different fusing options had to be taken into account.

These factors were discussed at length during the pre-Critical Design Review and Crew Station Working Group phase of the upgrade program. After much analysis and discussion with the user community, the aircraft System Program Office (SPO) and the operational community suspected that the developing interface concept might lead to degraded mission performance in certain circumstances and initiated a study to investigate this potential problem.

THE JDAM RE-DESIGNATION STUDY

A study using part-task and full-mission, human-in-the-loop (HITL) simulation was initiated and supported by the SPO and was conducted at Air Force Research Laboratory (see Figure A-5, the JDAM Re-designation Study). It sought to determine the impact of JDAM weapon re-designation on mission performance. The main independent variables -- or study factors -- were the difficulty of the re-designation problem, and the weapon load. The dependent measures included the time it took the WSO to re-designate all the weapons during a mission, the accuracy with which he performed the procedure, the number of targets attacked, and workload and situation awareness measures.

The difficulty of the re-designation problems was established by the number of decisions that needed to be made during the re-designation procedure -- as well as the complexity of the data manipulation that needed to occur in order to arrive at those decisions. Large numbers of decisions and / or complex data manipulations generated high difficulty problems, whereas small numbers of decisions and / or simple data manipulations generated lower difficulty problems.

Weapon load varied between loads of a single type of bomb body to two different types of bomb bodies.

Initially, operational WSOs participated in a part-task simulation of re-designation screen operation to characterize performance at the task level. Only re-designation time and accuracy data were collected during these part-task simulations. Subsequent full-mission simulation was conducted with a subset of the same WSOs to identify the re-designation task's impact on overall mission effectiveness. This approach enabled an understanding of the task-level performances of re-designation (such as re-designation time and accuracy), as well as an understanding of how those task-level performances translated into negative impacts (if any) on mission performance.

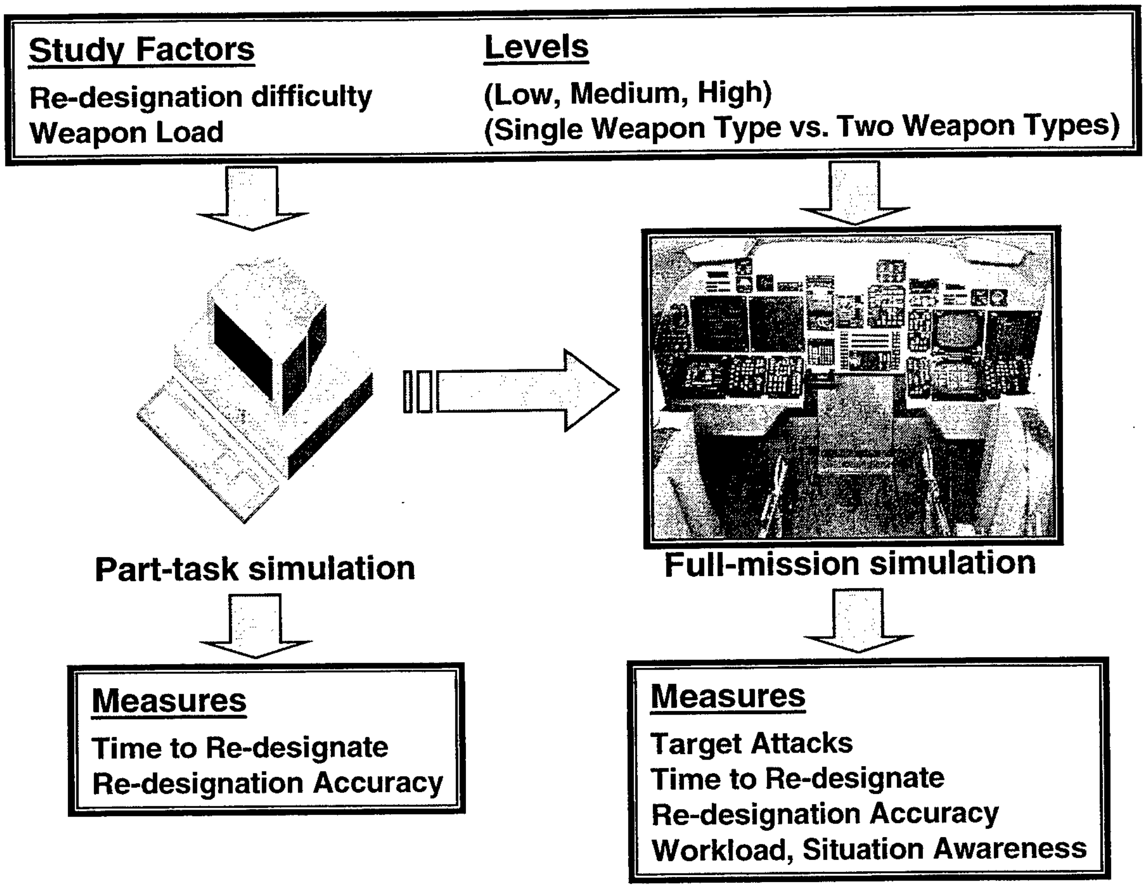
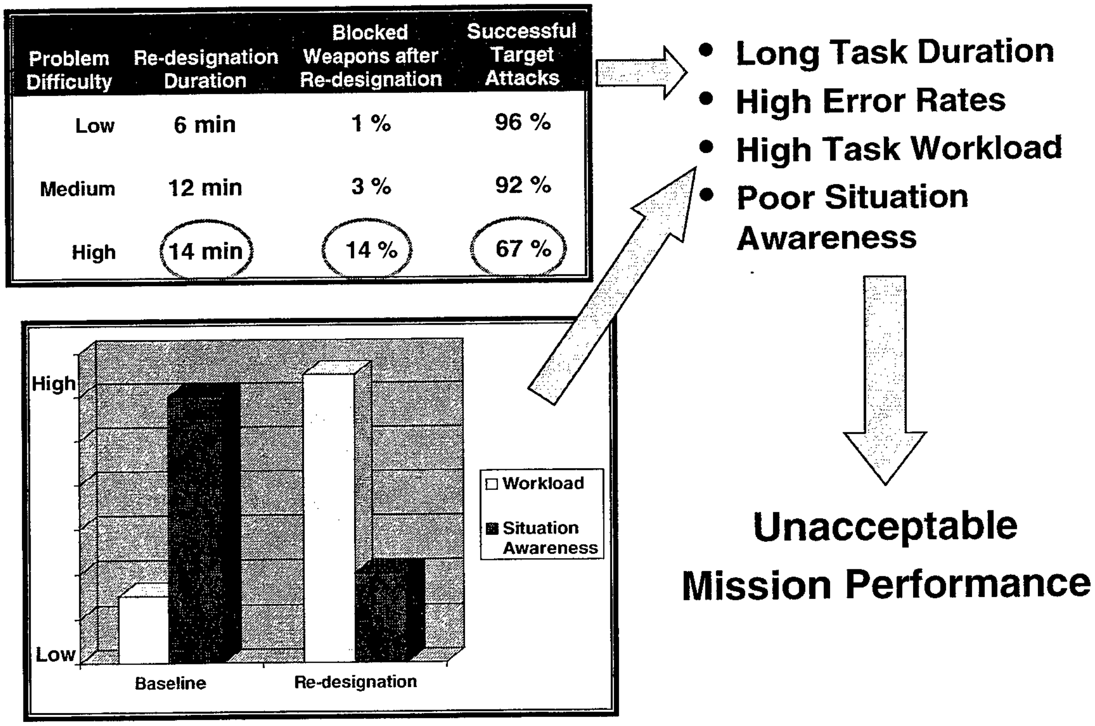


Figure A-5. The JDAM Re-designation Study.

Study Results

The results of the study highlighted the deficiencies of the crew system interface used for the JDAM re-designation task (see Figure A-6). For high-difficulty re-designation tasks, the average time to re-designate was 14 minutes and approximately 14% of the weapons remained logically blocked upon completion of the weapon re-designation task.

At the mission level, errors in re-designation performance reduced successful target attacks by as much as 33%. The difficulty of the task generated errors in weapon re-designation that prevented as many as one-third of all weapons from being released against an appropriate target.



Problem Difficulty	Re-designation Duration	Blocked Weapons after Re-designation	Successful Target Attacks
Low	6 min	1 %	96 %
Medium	12 min	3 %	92 %
High	14 min	14 %	67 %

- **Long Task Duration**
- **High Error Rates**
- **High Task Workload**
- **Poor Situation Awareness**

Unacceptable Mission Performance

Figure A-6. Results of the JDAM Re-designation Study.

In addition, WSO workload was reported to be very high when performing the target re-designation task when compared to baseline mission activities. Most often, this high workload was evident in the WSO being completely engrossed in the re-designation task to the exclusion of all other tasks -- such as navigation and mission time management. During this time, the Offensive Systems WSO shed tasking by passing navigation duties to the Defensive Systems

WSO, potentially putting the aircraft at increased risk to threats. Subjective comments from WSOs indicated that their situation awareness decreased significantly while performing the re-designation task, which would be expected given the previous task shedding observation.

Impact on the JDAM Acquisition

Given study results, it was determined that the initial JDAM interface concept required implementation changes in order to satisfy mission flexibility requirements.

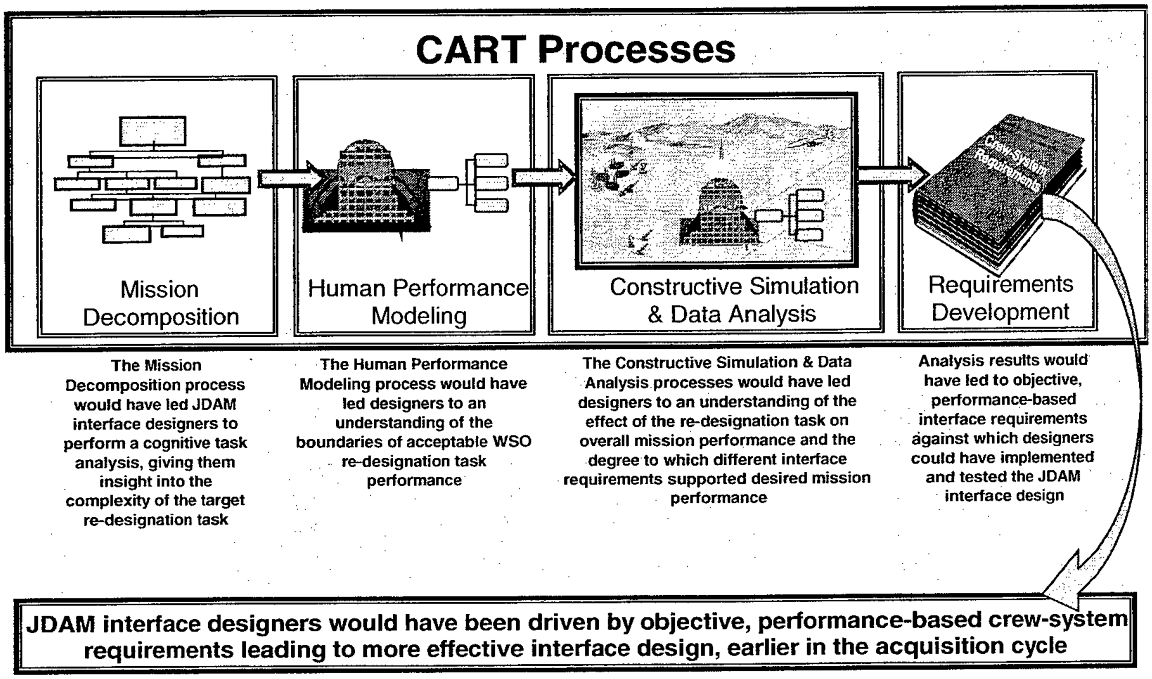
There were two major factors, which contributed to the redesign effort. As with most major acquisition programs, cost, schedule, and manning constraints precluded a detailed cognitive task analysis. This analysis would have provided deeper insights into mission complexity, and would have revealed the difficulty of the re-designation process and the demands placed on the operator by this task as a result of the interface concept. A thorough mission decomposition of the re-designation process would have revealed the complexity of the weapon-target re-designation task given the demands of limited time. Second (and more importantly), interface designers did not have any quantifiable subsystem requirements that specified -- in terms of objective performance criteria -- the overall mission performance that the interface needed to support.

As is the case in situations where quantitative performance-based requirements are lacking, the design was driven by other constraints -- such as minimizing changes to existing hardware and software. Over two block upgrade programs, the changes that were needed to adequately satisfy mission requirements were made to the WSO's interface. This was accomplished using human-in the-loop studies and simulations collecting such data as re-designation duration times, number of blocked weapons after re-designation, and successful target attacks. The new interface concept was eventually accepted by the user. Had the CART technology been available at the outset of interface concept development, multiple interface designs could have been evaluated in terms of their effect on mission performance, and later human-in-the-loop simulations could have concentrated on the most promising few prior to CDR.

CONCLUSION

The example presented here represents a typical acquisition scenario wherein a more exhaustive consideration of crewmembers during the development of system requirements and early design phase would benefit the program. Unfortunately, for this and other programs, CART did not exist to help solve this fundamental problem. Future acquisition programs, however -- especially those that are simulation-based -- need not suffer from these common pitfalls.

To accomplish the goal of properly considering crewmember capabilities and limitations in system design, CART incorporates a process that consists of front-end analytic activities, human performance modeling activities, and data collection and analysis activities. Figure A-7 illustrates how following that disciplined process would have resulted in effective requirements generation and design.



The Mission Decomposition process would have led JDAM interface designers to perform a cognitive task analysis, giving them insight into the complexity of the target re-designation task

The Human Performance Modeling process would have led designers to an understanding of the boundaries of acceptable WSO re-designation task performance

The Constructive Simulation & Data Analysis processes would have led designers to an understanding of the effect of the re-designation task on overall mission performance and the degree to which different interface requirements supported desired mission performance

Analysis results would have led to objective, performance-based interface requirements against which designers could have implemented and tested the JDAM interface design

JDAM interface designers would have been driven by objective, performance-based crew-system requirements leading to more effective interface design, earlier in the acquisition cycle

Figure A-7. How CART Could Have Ensured a Successful JDAM Interface

The front-end analysis is composed of a mission and task decomposition, which, in turn, includes an analysis of the information needs of the operator. A mission / task decomposition likely would have revealed the cognitive complexity of the re-designation process and would have provided a more complete understanding of operator information needs, mission requirements and their links to the system constraints.

Human performance modeling, coupled with the data collection and analysis activities, would have enabled analysts to predict the operator performance on the re-designation task and its resulting impact on mission effectiveness -- while reducing the need for costly, iterative human-in-the-loop simulation. For example, the study that was conducted determined that a 14-minute time-to-re-designate was too long to effectively support mission flexibility requirements. However, the study did not identify what may have been the acceptable performance limit. Simulations using CART could have established what the performance limits should have been, in terms of accuracy and latency. These limits could then have been used as performance criteria against which to design and test the crew system interface. Ultimately, interface designers would have had more objective, performance-based subsystem requirements against which to design their interfaces. This is a critical component missing from today's analyses, which when used with subjective feedback, provides ample evidence of a crew station's impact on the system's effectiveness and usability.

It is important to realize, however, that this situation is typical of past and current interface design. Lack of clear, objective, performance-based interface requirements have hampered crew station designers for decades. Because there has always been difficulty with quantifying crew system interface requirements from a human performance perspective, many interface requirements have been restatements of design constraints instead of statements of performance goals.