2 ABOUT THE CCRTS

2006 CCRTS
THE STATE OF THE ART AND THE STATE OF THE PRACTICE

HCI Design Patterns for C2: A Vision for a DoD Design Reference Library

Terry Stanard, PhD, Jeff Wampler, & Kendall Conrad
Air Force Research Laboratory
Cognitive Systems Branch
(AFRL/HECS)

Glenn Osga, PhD
Space and Naval Warfare Systems Center San Diego
User-Centered Design

POC: Terry Stanard
Warfighter Interface Division, AFRL/HECS
2698 G Street
Wright Patterson, AFB 45433-7604
(937) 255-9938 (Voice), 255-4250 (Fax)
terry.stanard@wpafb.af.mil

This document is cleared for Public Release by AFRL Public Affairs.
Document Number AFRL-WS 06-0107

Report Documentation Page

Form Approved
OMB No. 0704-0188

Public reporting burden for the collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to a penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.

1. REPORT DATE **JUN 2006**	2. REPORT TYPE	3. DATES COVERED **00-00-2006 to 00-00-2006**

4. TITLE AND SUBTITLE **HCI Design Patterns for C2: A Vision for a DoD Design Reference Library**	5a. CONTRACT NUMBER
	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **Air Force Research Laboratory,AFRL/HECS,2698 G Street,Wright-Patterson AFB,OH,45433-7604**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES
The original document contains color images.

14. ABSTRACT

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**		**34**	

ABSTRACT

Command and Control (C2) operators require well designed human computer interfaces (HCI) to effectively perform cognitive work. However, a methodology for transforming a requirements analysis into a useful HCI design is lacking. HCI Design Patterns (HCI DP) may help bridge this “design gap”. A set of reusable patterns known to support work functions could reduce the cost and risk associated with the design of future systems. HCI DP are an offshoot of architectural design patterns, used to catalog architectural solutions for recurring architectural design problems. Libraries of HCI DP are viewable online, but they commonly assist user interactions with generic system functions rather than actual C2 work activities. The Air Force and Navy are identifying HCI DP to assist the cognitive and collaborative work of C2 operations. Objectives include 1) Reverse engineering existing HCI designs and indexing them via cognitive work functions, 2) Developing a HCI prototyping environment embedding design patterns and indexing systems. A DOD-wide HCI DP library could promote a new set of HCI standards across the services. Future designs using a common set of patterns will promote interoperability between operators in different armed services collaborating on joint missions.

INTRODUCTION

Net-Centric Architectures and the HCI

Technology advances have created new opportunities and challenges for today’s warfighters, particularly those in Command and Control (C2) centers. In previous years information technology (IT) assisted C2 operators with selective parts of their work such as processing sensor data communicated over a small number of limited-bandwidth channels. Net-centric architectures are changing this. Centralized data warehouses with web-accessible services make data once available to only a selective few individuals now accessible by virtually anyone, anywhere with a computer, a wide area network connection, and access credentials. Net-centric architectures increase the quantity and variety of information many times over, and also greatly expand the network of the interconnected individuals, organizations, and systems. As a result, C2 operators will become increasingly pre-occupied with leveraging and managing this information universe, as well as the expanded human and technology network. As net-centric technologies become the norm, cognitive and collaborative tasks of C2 such as planning, decision making, building and maintaining situation awareness, and coordinating with others will be mediated near exclusively by IT.

C2 operator reliance on IT to perform their duties makes the Human Computer Interface (HCI) a key element in the success or failure of the net-centric paradigm. If operator performance is essential to accomplishing a C2 mission, and the HCI is the primary tool for the operator to engage the net-centric world, then it necessarily follows that the HCI is one of the most important facets of the total system architecture. The issue then becomes, what is the most effective way to design an HCI for a net-centric world? Do the old rules apply?

Although technology continues to evolve, what will likely to endure are the specific requirements of the user in the C2 context, what we call the work requirements. The work requirements are the specific goals, tasks, and natural operating modes of a C2 operator. A number of successful designs fielded within C2 centers are built upon a "work-centered" framework (1, 2, 3, 4). However, in today's fast-paced development world, a work-centered approach to HCI design must match the speed and agility of the concept-to-product design cycle inherent in web-based design. Thus, production of work-centered designs requires combining quality of design with speed and agility of approach. Selection of design methods that best balance these considerations is increasingly important in selecting a development process.

Three HCI Design Approaches

Any good IT project manager also wants to deliver a product that will be most useful to the end-user, that is, maximizing work-support. However, there are other considerations in managing IT projects. Cost reduction is an important goal. If time is money, then any successful design approach must keep pace with the speed and agility of the concept-to-product design cycle inherent in web-based design, which is the norm for net-centric architectures. Another goal is risk-reduction. Any IT project manager wants to minimize sources of uncertainty in meeting the project schedule and achieving key performance parameters (KPPs). IT project Cost, Risk, and Work-Support goals are always in tension, and different design strategies will achieve different balances across them. In this section we compare two contemporary approaches to designing HCI to balance these goals, and propose a third approach under development. As we shall see, these design approaches can complement one another. Before describing each approach, we have found it useful to ground the discussion of alternate design approaches within a practical story. The story contains metaphors that are useful in communicating the essential qualities and differences of the design strategies.

Costume Design Story

This past Halloween, the first author had to select a costume for his three-year old son. Asking the boy what he wanted to be, the reply was, "Darth Vader." The local department store had a ready-made Darth Vader costume hanging on a rack with an approximate fit. Further inspection at home found the costume a little big, but after some hemming and pinning, it fit reasonably well. But then the boy asked, "Daddy, what are you going to be?" A Jedi Night was a logical choice, but the store didn't have a ready-made Jedi costume. Star Trek costumes were available and might be "close-enough" to a Star Wars Jedi character for some people, but are clearly different for those with knowledge of the movies. Next, mom offered to watch the Star Wars movies and examine movie stills to get an idea for costume design, hoping to make a custom Jedi costume from scratch. While ambitious, concerns arose about the time to make the costume and confidence in what the finished product would look like. A third option came to mind. Are there costume patterns available? Yes! A costume pattern for a Jedi knight was discovered at a local fabric store. Mom read the instructions on the packaging, but after arriving home, she found it difficult to understand the patterns. Help was provided by a friend with experience creating clothing from patterns, resulting in a costume that fit well and gave authentic appearance; a Halloween success story.

Off-the-Shelf. This story about costume design is useful for illustrating three different approaches to designing an HCI. Figure 1 shows the comparisons. The first approach, "Off-the-Shelf," represents the way many or most HCI are designed for complex environments like C2. The HCI designer looks for an existing HCI product to suit the work domain. In some cases, there may be a ready-made HCI that suits the work requirements well. It's like finding a ready-made Darth Vader costume at a department store. However, there are often cases where there isn't a finished HCI that adequately supports the cognitive and collaborative work, so one is selected and "made to fit," albeit not very well. It's like selecting a Star Trek costume when what is really needed is a Star Wars costume.

A real-world example of an "Off-the-Shelf" approach is the Integrated Management Tool, an HCI designed for Flight Managers within the Tanker Airlift Control Center (4). It borrows the existing "Interface Idiom" (5) of a Microsoft Excel Spreadsheet. The information used for Flight Management tasks is presented within individual cells of a spreadsheet. There are over 80 fields or columns of information. Some additional programming collects and updates the information in near real time, and generates visual alerts in individual cells when a business rule is violated. All of the information needed is present, but the actual tasks associated with Flight Management require additional tools and strategies the adapted Spreadsheet does not offer.

The advantage of the "Off the Shelf" approach is low-cost and lower software implementation risk. The Excel Spreadsheet has years of development and fielding behind it. The user interaction protocols are established and accepted by users. It is therefore relatively inexpensive to adapt Excel for the Flight Manager HCI. Program management risk is also low since Excel is stable and predictable. The disadvantage, however, is the HCI is not tailored specifically for the Flight Manager work. The foundational HCI idioms, representations, and concepts do not reflect the way Flight Managers think about and conduct their work. They have additional cognitive overhead in making use of the information provided by the Integrated Management Tool. Thus, the HCI design succeeds on cost and implementation risk, but falls short in actual user work support.

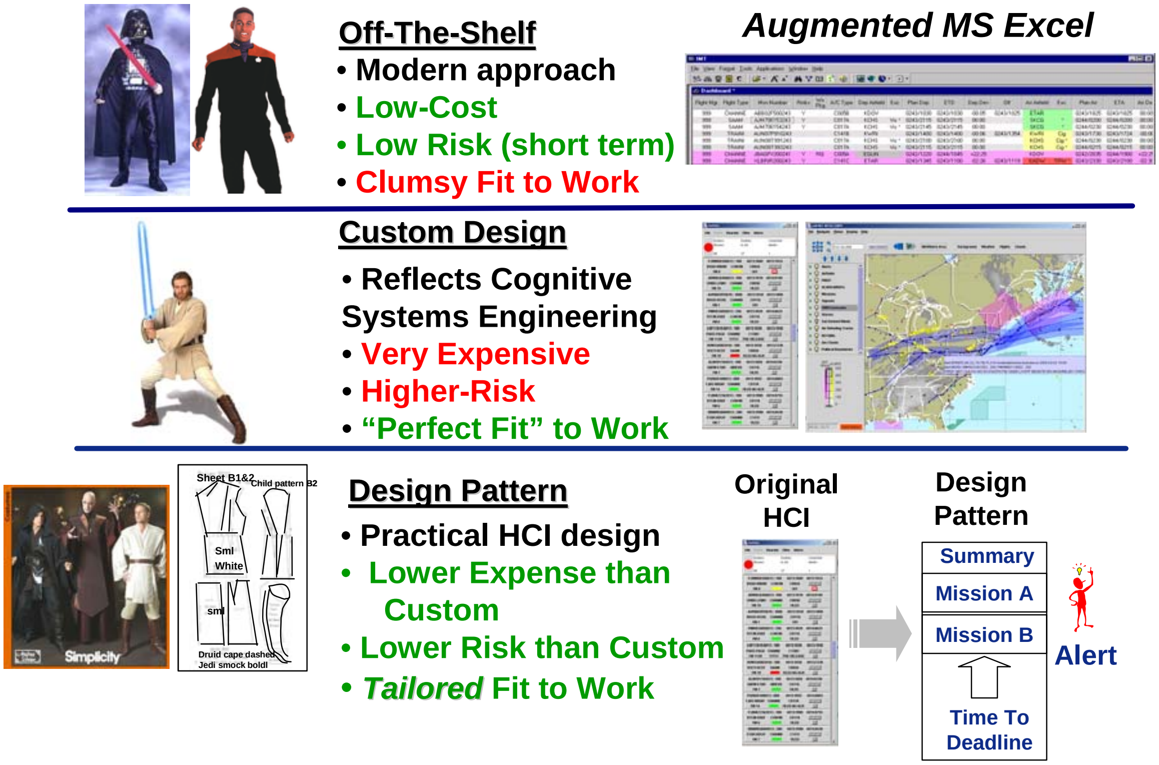


Figure 1. Three Approaches to Designing HCI for Complex Environments

Custom Design. A second approach is Custom Design of the HCI. Rather than rely upon an existing HCI for inspiration, one is developed from scratch to uniquely fit the work requirements of the user group and operational domain. The finished product is a custom, one-of-a-kind solution, analogous to creating a Jedi costume anew by watching the Star Wars movies. This approach reflects the current practices of Cognitive Systems Engineering (CSE). CSE is an HCI design discipline rooted in cognitive psychology and complex systems theory (6). There are several variants, but all reflect a firm commitment to understanding and supporting work-as-practiced within the actual operational context. The analysis phase often requires many months or even years for CSE practitioners to fully immerse in the domain and capture the cognitive requirements and complexities confronted by user groups. The prototype design phase is also lengthy. Thus, it could be said that CSE has no current repeatable methodology for transforming work requirements into a work-aiding HCI. It is a creative exercise leveraging the experience and innate abilities of the HCI designer working in concert with the CSE analyst. The absence of a repeatable methodology, also known as the "design gap," means significant time and effort is required to identify the appropriate representations and HCI supports for the work.

The key advantage of the Custom Design approach is the high level of work-support the results offer to the user/operator. Projects produced within CSE often receive very high acceptance among users, and if they undergo performance evaluation, often significantly improve performance of operators using the new HCI over a baseline HCI.

For example, high user acceptance and performance impact was achieved in a recent evaluation of a timeline scheduler developed to replace the Integrated Management Tool (Report Forthcoming). The disadvantage of the Custom Design approach is the additional time and money costs associated with the undertaking. Project management risk is also high, because the quality and performance attributes of the finished product are also virtually unknown at the outset of the project. This combination is often unacceptable to project managers balancing a thin budget and timeline, and who want some assurance of how well the finished HCI will perform and interface with other systems.

A design method is needed that captures the work-support benefits inherent in CSE custom designs, but that reduces risk and time and monetary costs. A complimentary approach to custom design could involve the abstraction of successful HCI design elements from custom designs for re-use in other design problems that have similar task and user characteristics. In this manner custom design can be supportive of the third design approach that is discussed as follows. A key element in the re-use of custom designs or abstraction of custom elements into usable patterns will be the judgment of similarity of the task and work characteristics from the previous to the current design problem. Similar to the comparative judgment between Star Wars and Star Trek!

Design Patterns. A third approach is based on the use of HCI Design Patterns (HCI DP). Essentially this is akin to Custom Design with the purpose of speeding the design process and re-capturing successful design elements into the new design. The HCI designer begins with a set of abstract HCI representations that have previously demonstrated a positive impact on the particular types of work performed by human operators within a similar complex system. The HCI representations are then tailored to the specific C2 task environment, accommodating specific users, information, and environmental factors. We can liken HCI DP to costume design patterns, which provide guidance to a clothier on what fabrics and pieces to cut and sew, but allow for tailoring to the specific body of the costume wearer. HCI DP may more effectively balance cost, risk, and work-support considerations than Off-The-Shelf and Custom Designs. Creating an HCI using HCI DP is potentially less costly and risky than a Custom Design, because the HCI is built upon an existing set of work-enhancing HCI concepts. A potential advantage of HCI DP over the Off-The-Shelf solutions is the re-use of HCI concepts and data representations that enhance certain cognitive and collaborative tasks within certain environmental conditions. Thus, HCI DP represents a pragmatic approach to HCI design for complex systems that addresses some important factors in the HCI design process.

The idea of applying design patterns to HCI creation is not entirely new. The next section traces the development of design patterns from use in physical architectures (i.e., buildings), to software development, and finally to HCI design. As the next section will illustrate, however, the majority of existing HCI DP were not engineered to assist cognition and collaboration within complex system environments like C2.

HCI DESIGN PATTERNS: HISTORY & STATE OF PRACTICE

Design Patterns emerged in the 1970s from the field of physical architecture, that is, buildings and urban landscapes. Christopher Alexander, the founder of the concept, described design patterns as architectural design components that could be combined to create workplace and community environments (7). Design patterns are proven solutions to recurring design problems and have an invariant character, descriptive components, and noted relationships with other design patterns. One purpose of design patterns is to facilitate communication between architects and users. Alexander advanced that good patterns could even be created by users as well as architects, a somewhat controversial claim. Another purpose of design patterns is to present an architectural solution that effectively balances a conflicting set of user interests and environmental constraints, which Alexander called "forces."

The software architecture community began to use patterns as a risk-reduction approach in the 1990's (8) and they continue to hold wide appeal. However, these patterns were meant for use exclusively by software designers to trade solutions to abstract software design problems. They are generally unintelligible to users of software. User interests were also not factored into the set of forces to be balanced by the design pattern. Borchers (9) presents an overview of increased pattern use and development for object-oriented code noting that pattern use in HCI "is hardly touched upon in this series". Borchers contends that Alexander's original intent of capturing the goodness of design for the user in the environment may not be fully implemented in software patterns to date. This gap between software patterns and user needs began to be recognized as the HCI design community picked up the design pattern approach in the mid-1990s. Although HCI DP do not make strong claims about improving communication between end-users and HCI designers, the community did recognize a need for HCI DP to balance forces that include user interests and goals. For instance, an example of a conflicting set of forces within an HCI design are a work-related need to scan multiple sets of information vs. screen clutter vs. workload related to retrieving hidden information vs. screen size and workspace layout. The HCI design pattern community has worked towards establishing purposes and goals for pattern design and use. For example Bayle et al. (15) describe the results of a HCI design pattern workshop, where participants identified different uses of patterns:

> Capture and Description: describing key characteristics of a situation or event in a context-sensitive way;
> Generalization: generalize across varying situations, yet retain a certain concreteness;
> Prescriptive: Patterns (i.e. design patterns) can be used to prescribe solutions to commonly encountered problems in a particular design domain. Thus, patterns might be used as a way of presenting HCI guidelines, or guidelines for organizational design.
> Rhetorical: The concreteness of patterns, and the fact that they are drawn from the situation for which design is being done, makes them very appropriate as a *lingua franca*, a common way of talking about design issues that is accessible to designers (of whatever disciplinary backgrounds) and the users and inhabitants of the situation.
> Predictive: Patterns can provide a 'what-if' mechanism for reflecting on the possible impact of changes to a place or situation.

Web design has become a core application area for HCI DP. The existing "library" of patterns is oriented towards general-purpose human-computer interaction tasks. Current patterns are not yet useful for C2 applications beyond a simple and narrow set of tasks. There exist two main challenges for C2 system applications. First, acceptable taxonomies describing the work environment need to be created to provide 'anchor points' for pattern use. The taxonomy must be operationally related to the tasks performed in current or proposed systems, and must be understood by designers such that task situations can be matched to the design patterns. Second, acceptable patterns need to be created and tested that meet rigorous requirements for design quality with respect to human performance. Many existing patterns are there because they exist in current designs but not because they are known to enhance performance. The following text describes ways in which patterns may begin to be formalized for use by C2 HCI designers.

Two Types of HCI Design Patterns (HCI DP)

HCI DP for Information Technology Domain Interactions.

Figure 2 shows a comparison of two domains of interaction an HCI DP can support. At the front is the Information Technology Domain. HCI DP have been created to streamline interactions of a user with an IT system; to improve interoperability between the human user and IT infrastructure. These IT-Centered HCI DPs, as we call them, have been identified for the IT classes of websites, desktop applications, and mobile phones. The majority of existing HCI DP available today are IT-Centered, and do not directly support the work of users in an operational context.

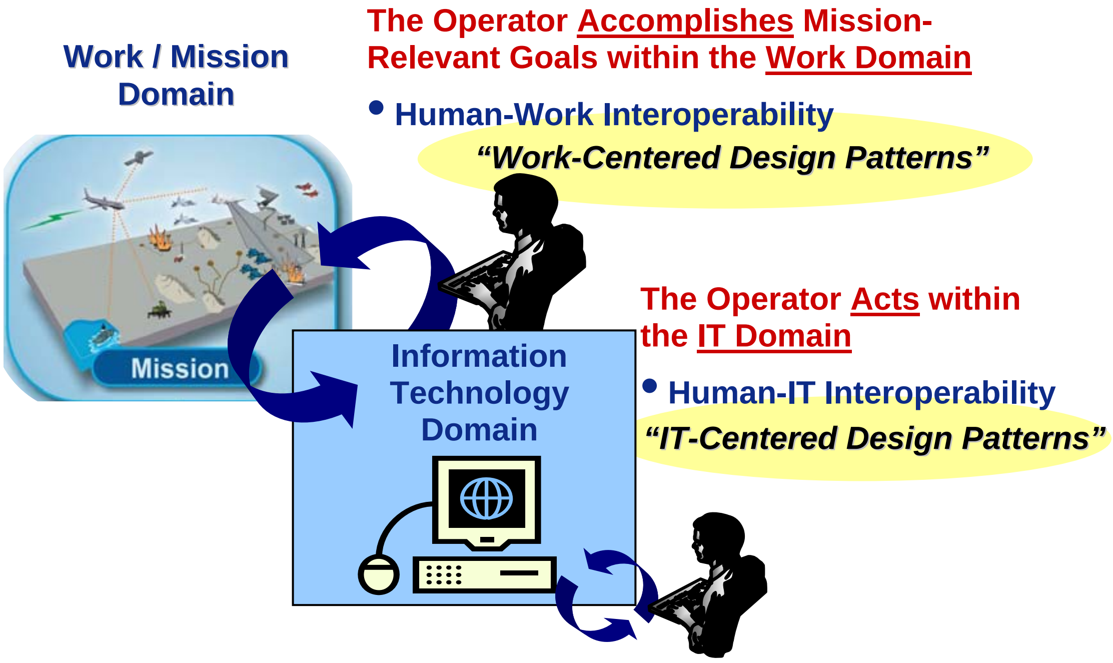
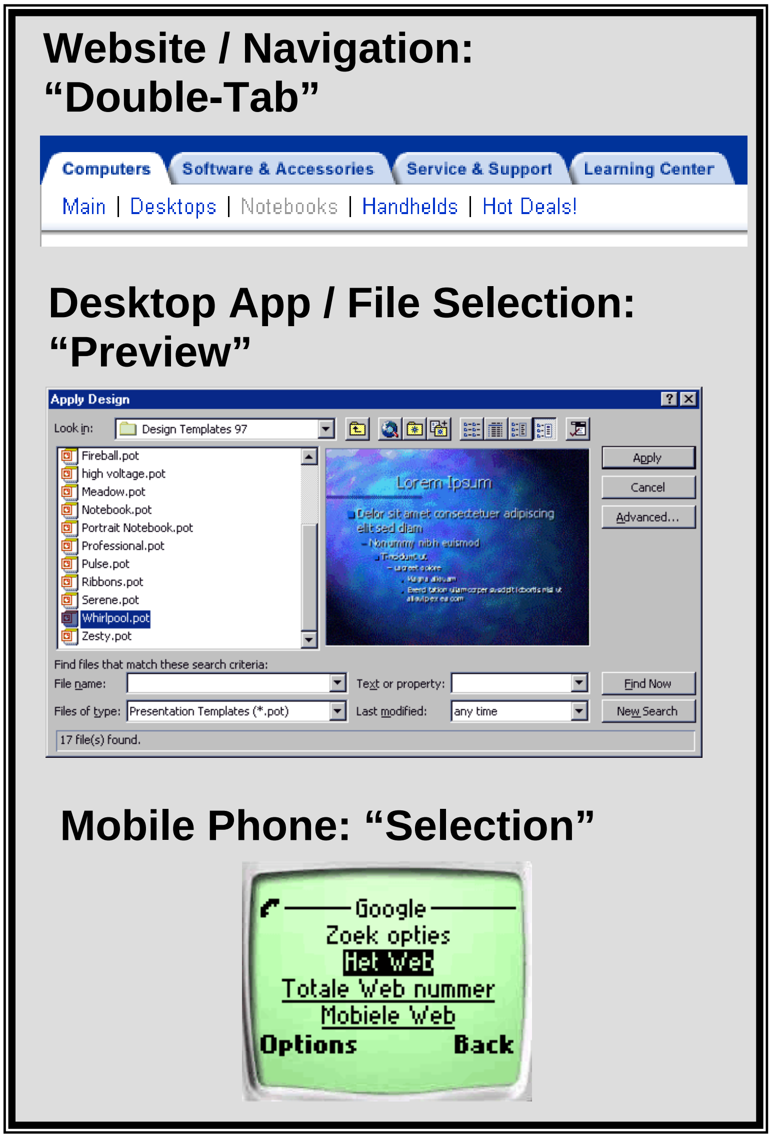

Work / Mission Domain
The Operator Accomplishes Mission-Relevant Goals within the Work Domain
• Human-Work Interoperability
"Work-Centered Design Patterns"
Mission
Information Technology Domain
The Operator Acts within the IT Domain
• Human-IT Interoperability
"IT-Centered Design Patterns"

Figure 2. Two domains of user interaction.

Figure 3 shows examples of three IT-Centered DP for three classes of information systems, all borrowed from the Welie online library (11). The first example, called “Double-Tab”, is for website interactions. The interaction task or “Problem” that motivates the design is navigating a website. The pattern solution leverages a file drawer metaphor, with the “folder tabs” representing the different content areas of websites. The second layer of tabs are a lower level of organization of website content. The second example, called “Preview,” is for desktop applications and the problem of selecting a file. The user is presented with a list of files, and is able to view a snapshot of the file contents before actually opening or executing the file. The third example, also called “Selection,” is for mobile phones and the problem of selecting a function or category of information to access. The user is presented with a list of options, and uses arrow keys on the phone to highlight different items. Once the item of interest is highlighted, the user then pushes a hard button to engage it.



Website / Navigation: “Double-Tab”

Desktop App / File Selection: “Preview”

Apply Design

Look in: Design Templates 97

Fireball.pot
high voltage.pot
Meadow.pot
Notebook.pot
Portrait Notebook.pot
Professional.pot
Pulse.pot
Ribbons.pot
Serene.pot
Whirlpool.pot
Zesty.pot

Find files that match these search criteria:

File name:

Text or property:

Files of type: Presentation Templates (*.pot)

Last modified: any time

17 file(s) found.

Mobile Phone: “Selection”

Figure 3. Three information system level HCI DP, drawn from http://www.welie.com

IT-Centered DP are essentially usability principles embedded within objects and dialogs. Some may even call them the next generation of widgets. Because these conventions are in widespread use, they reduce the learning curve of a user with a new system. But if employed within a C2 center, they would only indirectly help an operator accomplish mission-relevant tasks and goals. While assisting a user in interacting with a system, the user still has to accomplish their mission-relevant work. The patterns do not enhance interaction with the work domain, necessarily. They also are inadequate guides for HCI designers to follow in creating a useful design for C2 centers. It is entirely possible to build an HCI based on these patterns that is totally useless to a C2 operator. They are based on tasks to interact with the information system, and are not based on the tasks and goals of an operational domain like C2.

HCI DP for Work Domain Interactions

Returning to Figure 2, another domain of user interaction is the field of work or mission space. This is where the C2 operator ultimately directs their activities. The C2 operator uses IT to accomplish mission-relevant purposes and goals. HCI DP are needed that improve the operator's understanding of and interaction with the work domain (12.13). Work-centered HCI DP will enhance an operator's skill in performing tasks within the C2 operational area. The motivating tasks or problems are cognitive and collaborative in nature, and the forces the pattern resolves are competing goals and constraints that exist within the work domain. Appendices A & B provide examples of two Work-Centered DP.

Table 1 compares some of the activities associated with IT-Centered interactions and Work-Centered interactions (based on 14). Most existing HCI DP libraries can be linked with various kinds of IT-Centered interactions, and few with the categories of Work-Centered interactions. Thus, additional HCI DP are also needed for work interactions like those listed in the table. By way of comparison, Frank and Lillian Gilbreth identified a set of 17 standardized hand motions associated with manual labor, known as "Therbligs" (roughly, Gilbreth spelled backwards) (15). We endeavor to identify a stable set of fundamental cognitive and collaborative activities associated with the C2 work domain, and associate Work-Centered DP with these.

Table 1. Activities and interactions associated with IT Domain, and C2 Work Domain.

Information Technology Domain Interactions	**Work Domain Interactions**		
	Coordination	**Cognition**	**Behavior**
Organize Content	Share information	Monitoring	Control tasks
Organize virtual workspace	Build consensus	Storybuilding	Physical work space organization
Accept user commands, inputs	Distributed decision making	Detection & Recognizing events, objects, patterns	Constructing, assembling
Stylize content, virtual work space	Communicate intent	Hypothesizing	Material movement, transport
Navigating data, virtual space	Motivate and inspire	Perceptual Judgments	Skilled action
Selecting & Manipulating content, objects	Set goals	Inspection	Generating documentation
Search & Filter Data, Files	Handoff Work Products, Roles	Scheduling	Speaking/briefing
Save & Undo	Distributed Planning	Planning	Reading
	Brainstorming	Forecasting / Projecting	Operating equipment
	Relationship building	Diagnosis	Checking routines
	Negotiation	Simulating	
	Conflict resolution	Mental Simulation	
	Delegate	Course of Action Analysis	
	Monitor others workload, workflow	Time Management	
		Prioritizing	

Note: Based on Tables 25, 26, 34 in Whitaker et al., 2005)

VALIDATING HCI DESIGN PATTERNS

An important issue associated with design patterns is verifying their stability and suitability for solving the motivating problem. Alexander thought the best way to "validate" a design pattern was through the volume of cases where the pattern was applied. If the pattern is in wide use, this is evidence of its invariance as a reusable solution. Patterns with many real-world implementations have high validity, and those with fewer examples to substantiate them have lower validity. This same principle was carried forward into the HCI domain. The HCI DP in most existing libraries have relatively high validity, because there are many implementation examples. Implicitly, high case volume is an indicator of the usefulness of the pattern. We will call this **Validity through Invariance**. There are regrettably few HCI designed for C2 environments that can be called work-centered. As a result, work-centered HCI DP must be abstracted from individual, or small samples of finished, fielded work-centered HCI designs. Thus, Validity through Invariance will be difficult or impossible to satisfy in the Alexandrian sense for the foreseeable future. Appendices A & B feature Work-Centered DP culled from HCI designs created at AFRL and SPAWAR.

We have identified alternative bases for gauging HCI DP validity: Work Content Validity and Performance Validity. **Work Content Validity** means the content of the HCI DP corresponds with the work as practiced and experienced by the operator in the operational domain. The features and functions of some HCI DP will correspond to the work in some C2 contexts, but other HCI DP may not correspond well, and are not appropriate to apply. Work Content Validity is established through work analysis methodologies such as Cognitive Systems Engineering, which model the complexity of the operator's work in a real-world environment. Work Content Validity may be measured through domain expert ratings of the HCI DP. The latter gives some objective basis for Work Content Validity. **Performance Validity** means the HCI DP has been shown to demonstratively improve user performance on the target task(s). This validity type requires that performance measures be developed and the HCI DP experimentally tested against alternative or baseline HCI concepts in use within the domain of interest. It is not clear how many of the existing IT-Centered DPs have been empirically tested to evaluate their impact on speed of learning a new system, or for improving speed or error rate on the relevant interaction tasks.

DESCRIBING HCI DESIGN PATTERNS

An important issue with HCI DP is effectively communicating them so they may be reused. At a minimum, the pattern description should make clear the motivating problem, so an HCI designer will recognize if it corresponds to his/her design problem. The pattern must also prescribe a HCI solution that can be reliably adapted and put into practice. There are several ways that HCI DP are described in current libraries, some following an Alexandrian template that includes a number of prescribed information fields. Other versions do not adhere completely, or do not follow at all the Alexandrian approach (16). Our current bias favors an Alexandrian template, because of the range of information fields it includes. Table 2 below, adapted from (14), lists the most common fields or attributes of an Alexandrian template applied to HCI DP.

Table 2. Common attributes of a HCI DP description template

Pattern Attribute	**Description of Attribute**
Name	A concise and meaningful label for the pattern
Problem	A statement of the particular situation the HCI DP addresses.
Context	The setting where the problem and its solution occur.
Forces	A description of the relevant tensions between possibilities and constraints, how they interact and conflict, and how they relate to the problem.
Solution	A description or specification of an HCI solution to resolve the problem and the acting forces within the stated context.
Picture/Diagram	A summary graphic representation of the pattern and its components.
Validity	A measure or a means for assessing the "rightness" of the given pattern for the given context and problem. In practice, this is an index of the invariance of the pattern as demonstrated through repeated implementations.
Examples	One or more sample views of the HCI solution as implemented in deployed designs. Note the actual implementation may feature customization to suit the particulars of the context and creative expression of the HCI designer.
Smaller Patterns	Patterns that are constituents or components of the pattern of interest

An examination of the current HCI DP libraries reveals that most descriptions are not very information rich, constituting one notebook page or less of content. This brevity may reflect the relatively low level of complexity of user interactions with the IT domain. It may be unnecessary to create larger volumes of content to describe HCI DP for IT interactions. There is also an advantage with brief descriptions, in that they do not "bog down" a reader with too much information. The reader is able to more quickly apprehend the essence of the problem and solution, and decide whether or not it is applicable.

In cataloging work-centered HCI DP, we are finding it necessary to include more content. The focus is on interactions with the work or mission domain, and the high complexity of C2 environments would seem to require significant explication of the overall problem, acting forces, and operational context in which they are found. These aspects are described in terms of the goals to be achieved within the operational domain, cognitive and collaborative tasks and activities, and operational priorities and constraints. Similarly, the HCI solutions themselves are often more complex, requiring significant space to describe sufficiently. Introducing the fields of Work Content Validity and Performance Validity also introduces new description requirements, demanding at least a brief explanation of the methodology used to build an understanding of the work, the level of correspondence between the pattern and the work as practiced by seasoned operators, and the results of a work performance impact evaluation.

FUTURE VISION

Once the preliminary research associated with Work-centered HCI DP are complete, we can begin to envision many uses and benefits. The primary benefit of an HCI DP is the generalization of point solutions to the design community to allow rapid reuse and adaptation of existing proven solutions to future C2 applications. There is traditionally a huge cost and time associated with the design of complex HCIs for C2 systems. Every time a new capability becomes available, the government invests large funds to procure the new capability through development projects that begin from "scratch." The existence of a **Design Reference Library** of HCI DPs will allow analysts and designers to browse and match proven HCIs to the specific work requirements of their current design problem. The creativity and risk associated with design will be reduced to customizing the pattern for the specific domain of re-application, instead of creating the HCI anew. This approach will increase the likelihood of knowledge transfer from earlier, successful projects designs, while significantly reducing the cost.

Flexible, Reusable HCI Code

With the continual re-application of proven HCI solutions and refinement of the Design Reference Library, the HCI design community will ultimately converge on a limited set of C2 operator "work field" representations and interactions contained within robust HCI DP. The HCI DP will become a new set of HCI standards, with some flexible and some relatively inflexible parts. The work field representations and interactions will be common across similar C2 environments (inflexible), but the designer will selectively customize some features as s/he applies the pattern to the specific user population, their information requirements, and operational setting (flexible). The resulting design will thus retain the skeletal standard framework, composed of Work-Centered DPs and IT-Centered DPs.

HCI DP can exist on paper, and that is their current form. However, they could be even more useful if embodied as reusable software code. Instead of providing additional written design guidance, which could be viewed unfavorably by designers as additional overhead documentation, the guidance could be formally codified in constrained GUI forms. This could make a Design Reference Library much more work-centered from a developer point of view. For instance, design widgets are a common tool within an Integrated Development Environment used to build HCIs. Examples are radio buttons, pull-down menus, and so on. The HCI DP could become a next generation of IDE widgets, albeit more complex in nature. The HCI DP would exist as software modules, with configurable (flexible) components but more static (inflexible) components to maintain the integrity of the underlying pattern. The HCI designer would select HCI DPs from an electronic library, assemble them into a coherent interface, and populate them with content appropriate to the operational user group. Assistance would need to be provided to the designer to locate appropriate HCI DP, and we are currently considering several different "entry paths" or indexing systems for a Design Reference Library. There are many other methodical and technical challenges as well, but it is an exiting vision.

Joint Operations and Training

A common set of HCI DP embodied in a Design Reference Library and deployed across services and agencies will better support the prevailing conditions of joint operations, rotating assignments, and reduced manning. C2 operators are working more and more with other operators distributed geographically, as well as across services and agencies. Thus, "human to human interoperability" is a key enabler for a successful net-centric operation. As future HCI become standardized through the Design Reference Library, distributed operators working from HCI based on similar core set of HCI DP will find it easier to share information and effectively collaborate since their IT tolls give them complimentary views of their individual and common work space. In addition, duty officers routinely rotate through C2 centers, requiring training time for incoming personnel on the IT. A Design Reference Library featuring HCI DP that effectively mirror the cognitive work of C2 will reduce the training time necessary for rotating duty officers to become proficient with their IT toolsets and productive members of the organization. The present focus on reduced manning makes every operator a critical element of the C2 enterprise, and for this reason the speedy inception of new personnel is also important.

SUMMARY

> "Many interfaces are, and certainly appear to be, last-minute affairs, thrown together so that the users have something to interact with." (17, page 347)

Many IT projects for C2 centers produce HCI with a non-optimal allocation of functions between human operators and IT, and burdensome interaction protocols for human operators trying to perform their assigned tasks to achieve mission objectives. Although IT developers aim to produce useful tools for warfighters, many have limited training in human information processing and HCI design, and modify available, existing HCI concepts on new projects with limited understanding of their suitability for the cognitive and collaborative tasks of operators. As a result, many IT systems are produced at significant expense up front, but much greater long-term expense to the C2 center as new operators struggle to learn the system and seasoned operators struggle to find ways to make the IT support their work practices. The interoperability problems compound all the more within net centric operations, as operators interface with an increasing number of organizations, individuals, and systems, and manage a much greater variety and volume of information than ever before.

Cognitive systems engineering is a discipline devoted to producing HCI for complex environments like C2 centers that are effectively work-centered and more adequately leverage the unique capabilities of human actors and IT within the HCI design. However, while achieving the objectives of work support, CSE often fails to meet affordability and risk management goals. Lacking a repeatable design methodology, each CSE design artifact is fully custom, costing considerable time and money to produce, with little assurance for project managers and customers about the anticipated quality of the end product. As a result, CSE had not made the potential impact within modern C2 centers that it might.

This paper advocates a design pattern approach to HCI development to more effectively balance considerations of cost, risk, and work-support. Design patterns emerged from an architectural design movement beginning in the 1970s and were later adopted by software developers wishing to share best practices. More recently, design patterns have been adopted by the HCI design community to provide more actionable design guidelines. This community has produced catalogs of HCI DP supporting common interaction tasks of users with particular classes of IT, irrespective of the end-work the user is trying to accomplish. We call these IT-Centered DP. Additional efforts are needed to identify Work-Centered DP that support common operator work tasks within particular C2 domains of operations. A library of Work-Centered DP and IT-Centered DP can be leveraged by HCI designers to produce effective work supports in a more timely, cost-effective manner. There is even potential to create software code featuring the HCI DP, providing HCI designers with truly actionable guidance and meaningfully constrained developer tools. Additionally, HCI DP can establish a new set of HCI design standards across the DOD based on a design pattern approach, ensuring not only greater consistency in look and feel, but greater consistency of performance of operators within C2.

However, the promise of design patterns is accompanied by significant research challenges. Fundamentally, we lack a library of Work-Centered DP for C2. But a pre-requisite is taxonomy of cognitive and collaborative tasks for C2. This taxonomy is needed to properly index HCI DP, particularly those that are work-centered. The collective experience of CSE practitioners, subject matter experts, and existing work taxonomy such as Mission Essential Competencies, will help to distill the stable cognitive work tasks and context features associated with particular domains of operation. Also lacking, but with some initial progress reported here, is an adequate template for describing HCI DP, particularly those that are work-centered. The existing catalogs of IT-Centered DP do not all follow the same descriptive methodology, but more importantly, they are universally insufficient for describing the cognitive work problem and work context motivating a work-centered DP. Numerous challenges also await in transitioning HCI DP from a paper-form to software code.

Nevertheless, these challenges are exciting because they are directed towards solving real-world problems with clear significance for military operations, and national defense and security. These research topics are also exciting, because they are directed towards understanding fundamental patterns of work in C2 as well as the currently available HCI forms and interaction protocols that best suit those patterns of work. Investigating and cataloging fundamental patterns of work is not unlike the search for the fundamental forces and particles of matter and energy. We are commonly searching for the essence of something, in this case, complex cognitive activity as it emerges in the C2 domain.

REFERENCES

(1) Osga, G. (2003) Building Goal-Explicit Work Interface Systems. Paper presented at the *Human-Systems Integration Symposium, Enhancing Human Performance in Naval and Joint Environments,* June, Tysons Corner VA.

(2) Osga , G. (2003) Work-Centered Computing: Future Challenges: Paper Presented at the *9th Annual Human Factors & Ergonomics Symposium, San Diego Chapter*, Human Factors & Ergonomics Society. March, San Diego, CA.

(3) Scott, R., Roth, E. M., Deutsch, S. E., Malchiodi, E., Kazmierczak, T., Eggleston, R. G., Kuper, S. R., & Whitaker, R. (2005). Work-centered support systems: A human-centered approach to intelligent system design. *IEEE Intelligent Sys*tems, *20, 73-81*.

(4) Wampler, J.L., Whitaker, R., Roth, E., Scott, R., Stilson, M.,Thomas-Meyers, G. (2005). Cognitive work aids for C2 planning: Actionable information to support operational decision making. *Proceedings of the10th International C2 Research and Technology Symposium: The Future of C2*, McLean, Virginia, June 2005.

(5) Tidwell, J. (2006). *Designing interfaces: Patterns for effective interaction design*. O'Reilly Media: Sebastopol, CA.

(6) Eggleston, R.G. (2002). Cognitive systems engineering at 20-something: Where do we stand? In M.D. McNeese & M. Vidulich (Eds.), *Cognitive Systems Engineering in Military Aviation Environments: Avoiding Cogminutia Fragmentosa*. Wright-Patterson Air Force Base, OH: HSIAC Press, 15-78.

(7) Alexander, C., Ishikawa, S., Silverstein, M., Jacobson, M., Fiksdahl-King, I. & Angel, S. (1977). *A Pattern Language: Towns, Buildings, Construction.* Oxford University Press.

(8) Gamma, E., Helm, R., Johnson, R., & Vlissides, J. (1994). Design patterns: Elements of reusable object-oriented software. Addison-Wesley: Reading, MA.

(9) Borchers, J. (2001). *A Pattern Approach to Interaction Design*. John Wiley & Sons: New York.

(10) Bayle, E., Bellamy, R., Casaday, G., Erickson, T., Fincher, S., Grinter, B., Gross, B., Lehder, D., Marmolin, H., Moore, B., Potts, C., Skousen, G., Thomas, J. (1998). Putting it all together: Towards a pattern language for interface design. *SIGCHI Bulletin, 30* (1). Retrieved March 20, 2005 from http://www.sigchi.org/bulletin/1998.1/erickson.html

(11) VanWelie, M. (2005). Interaction Design Patterns. Retrieved August, 2005, from http://www.welie.com/patterns/index.html

(12) Stanard, T., & Wampler, J. (2005). Work-centered HCI design patterns. *Proceedings of INTERACT 2005: Communicating Naturally through Computers*, Rome, Italy, September 2005.

(13) Osga, G. A. (2004). Building task-product-focused user interfaces using information sets and design patterns. Proceedings of the Human Factors and Ergonomics Society 48th Annual Meeting, pp. 2416-2420. Human Factors & Ergonomics Society.

(14) Whitaker, R., Scott, R., Roth, E., Militello, L.G., Stilson, M.T., Wamper, J.L., Stanard, T.W., Thomas-Meyers, G.F., & Schmidt, V.A. (2005). Work-Centered Technology Development (WTD). Air Force Research Laboratory Technical Report #AFRL-HE-WP-TR-2002-07.

(15) Wikipedia (2006). Therbligs. Retrieved March 20, 2006, from http://en.wikipedia.org/wiki/Therblig

(16) Fincher, S. The Pattern Gallery. Retrieved March 20, 2006, from http://www.cs.kent.ac.uk/people/staff/saf/patterns/gallery.html

(17) Hackos, J.T., & Redish, J.C. (1998). *User and task analysis for interface design*. John Wiley & Sons: New York.

INTRODUCTION TO APPENDICES

Appendices A and B includes examples of two Work-Centered DP inspired by design projects originating from AFRL and SPAWAR. Our present approach to identifying Work-Centered DP is to reverse engineer fully-functional and fielded HCI and mine them for patterns. Functional HCI are complex, and we typically identify ***multiple*** IT-Centered and Work-Centered DP in a full design, some nested within others. The examples in Appendices A and B are ***complex, composite*** Work-Centered DP composed of smaller, constituent DP. Additionally, we are aware there are DP associated with the visualizations of the operator's work field, DP associated with the Alerting functions of the original HCI, and DP associated with the User Interaction modes. These HCI DP categories are not discussed in the body of this paper, nor explicitly labeled in the examples of Appendices A and B. Rather than label the DP constituents and DP types separately, we have included them all in the composite pattern description. Nevertheless, the careful reader will see where some of the discriminations can be naturally made. Our methodology for describing composite and constituent DP is evolving, and we will make the most recent versions of these pattern descriptions available at time of the conference.

APPENDIX A

Work-Centered HCI Design Pattern: **Mission Timeline Monitor (v 1.2)**

Work Summary

1. <u>Problem Statement</u>: User or work group needs to schedule an intended mission, or monitor the temporal unfolding of a mission.

- ➢ Examples: Scheduling aircraft flights, delivery shipments, phased business meetings

- ❖ Each mission consists of one or more required events, resources, and factors influencing timing of events and resource uses.
- ❖ Successful scheduling requires balancing conflicting priorities and demands associated with events, resources, and factors
- ❖ Challenges can include
  - ➢ Maintaining accurate and timely understanding of the mission status
  - ➢ Significant cognitive and procedural burdens involved in evaluating the viability status of a mission
  - ➢ High risk of information overload in scheduling and monitoring a mission
  - ➢ High risk of errors and oversights in scheduling and monitoring a mission
  - ➢ Coordinating among different actors tasked with different aspects of the mission

2. Context: The work context normally includes…

- **Mission**: A set of objectives or end-states that are achieved through a required set of events using a limited set of resources.
  - Examples of missions from a variety of domains are: airlifts, professional conference, shipping order
- **Events**: Required actions or phenomena that take place during the mission, occurring at a specific point in time or over a duration. Events are dependent on resources, and factors.
  - Examples within airlift missions: Takeoff, landing, refueling
- **Resources**: Specific items or elements associated with a mission, or associated with the context of the mission.
  - Examples within airlift missions: aircraft availability, designated cargo, aircrew availability, flight plan, overflight clearance duration
- **Factors**: Affect viability of resources or events within a mission. Essentially, business rules and laws associated with resources and events. Some factors are associated with resources, and others are associated with events.
  - Examples within airlift missions: crew rest periods, airfield operating hours, time to refuel.

The work context can include…

- A limited time window available for the mission
- Multiple actors deal with different aspects of the mission
- Multiple actors coordinate and collaborate to process a mission
- Multiple actors use different IT resources, data, and visualizations

3. Forces: Conflicting goals or influences include…

- Schedule the mission within a limited time window
  vs. Satisfy the many mission events, resources, and factors, all with temporal impacts
- Balance the maximum number of mission factors
  vs. Simplify or Reduce the number of interactions among factors to evaluate
- Monitor mission as a whole vs. Monitor individual facets of the mission

Solution Summary

1. HCI Description

A. Intent: The HCI solution leverages a Timeline Idiom to assist user tracking status of overall mission, as well as associated events and resources, in the time domain. The HCI solution also includes automated alerts to inform the user when the events and resource-uses violate business rules or factors associated with the mission.

B. Work Field Representation & Schematic – Visualizations of the operator's work space

i. Central Components – Portions of the work field representation that are centrally positioned, to draw immediate attention of operator

- Dimensional Space: Time, but not 3D space.
- Representational Format: 2D stack of horizontal rows
  - Horizontal axis corresponds to continuous time.
  - Vertical axis is divided into rows, each row assigned to the mission schedule, its resources or factors.

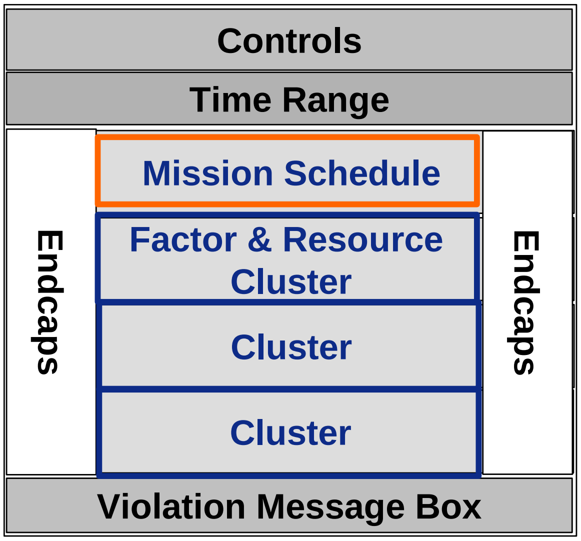


Figure A1. General visual layout of the Mission Timeline Monitor DP. Central Components are colored, and Peripheral Components are black type.

- ❖ Major Representational Components

  - ➢ <u>Timeline Concept</u>

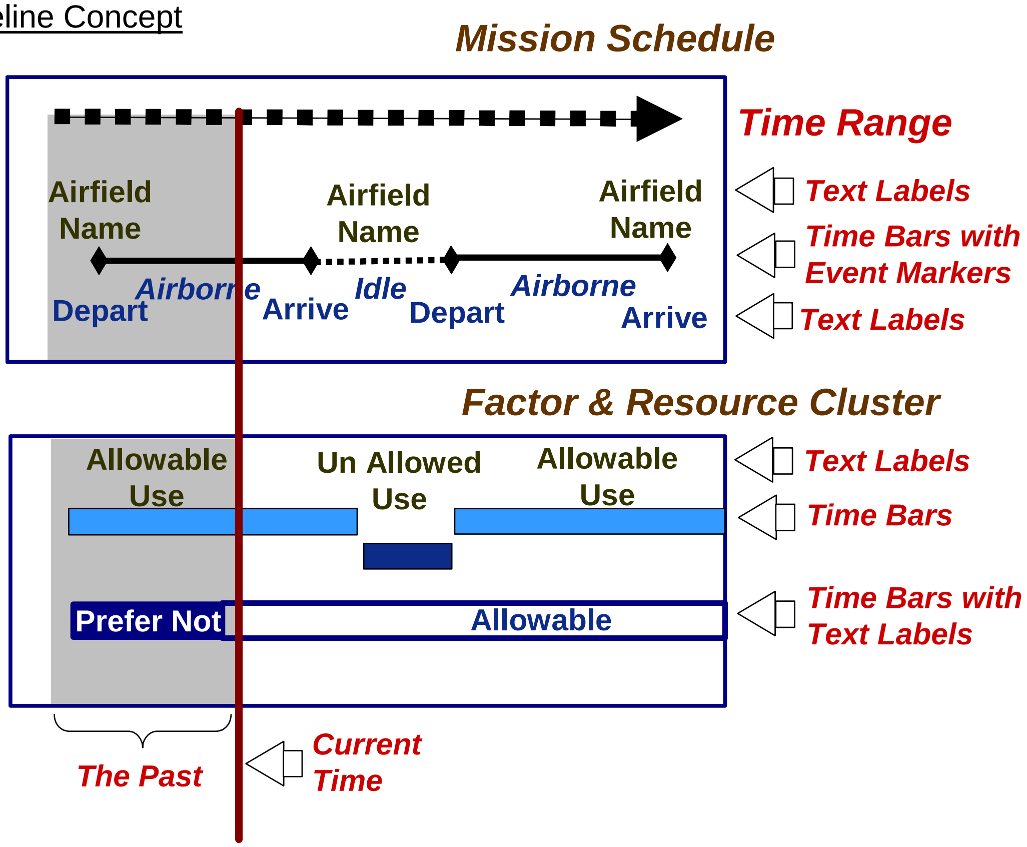


Figure A2. Major representational components of the timeline concept.

- **Time Range**: the temporal axis, represented as a horizontal bar at top or bottom of display frame, with tick marks corresponding to desired level of granularity (e.g., dates, days, hours, minutes, seconds)
- **Event Markers** corresponding to the timing of an event, such as departure or arrival in the air mobile domain.
- **Time Bar** corresponding to the time and timing of an event, resource, or factor. All time bars are temporally aligned with the same time range to allow relative temporal comparisons between them. Thickness and line type (solid, dashed, dotted) are used to visually discriminate among several time bars in close proximity.
- **Current Time**: a vertical line aligned with the current time on the time range, and crossing all time bars. The area before the current time is colored to show users they cannot affect this time period.
- **Text Labels** may be positioned over time bars and markers to communicate additional information.

- **Mission Schedule**: Regarding flight missions, depicts the events, durations, airfields associated with the mission. The corresponding time bars, markers are grouped and positioned at the top of the stack, above all factor and resource clusters. The mission schedule is surrounded with a visual border to discriminate it.

- **Factor & Resource Clusters**: Depicts the temporal constraints on mission resources and factors. See Figure A3. Constraints may be “hard” (unavoidable) or “soft” (negotiable). Event markers and time bars are aligned with the Time Range and Mission Schedule allowing temporal comparisons of common occurrences across all clusters (For example, the Current Time vertical bar crosses time bars within the Mission Schedule and Factor & Resource Clusters at the same point in time) . Related resources and factors are visually grouped by placing event markers and time bars in proximity to each other. Examples of clusters from the air mobility domain are: Country Overflights & Clearances, Airfield, Aircrew, Weather, Departure Readiness. Each cluster is surrounded with a visual border. There may also be a Cluster Summary, representing a summation of all temporal impacts associated with the individual factors and resources within a cluster. Mouse Clicks on time bars within a cluster allows the user to view Cluster Summary only, or accompanied by all cluster details.

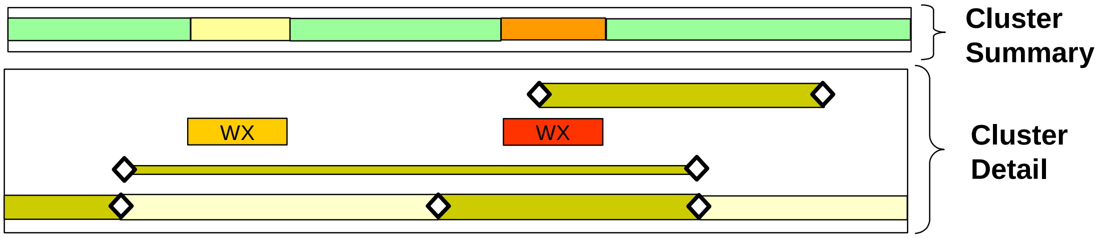


Figure A3. Illustration of Factor & Resource Cluster Summary, and Detailed views.

ii. Peripheral Components - Portions of the work field visualization that are peripherally positioned, allowing operator to shift attention towards them as needed, without cluttering the central components

- End Caps: Descriptive titles and labels placed on the ends of the Mission Schedule, and each Factor & Resource Cluster. A surrounding visual border separates the End Caps from associated time bars and event markers.
- Violation Message Box: A text box position below the Factor & Resource Clusters, where detailed messages regarding a constraint violation alert are presented.
- Controls: (See Interaction Controls below)

D. Contextualized Alerting – Intelligent agents monitor dynamic status of each mission event or resource availability for violations of factors (i.e., business rules or laws).

i. Central Components

- Time Bar Coloring: Color of time bars indicates allowable or unallowable timing or duration of events and resource uses
- End Cap Color Surrounds: Colored red when at least one time bar has violated a business rule.

ii. Peripheral Components

- Violation Message: A text message explaining the nature of the violation is displayed in the Violation Message Box.

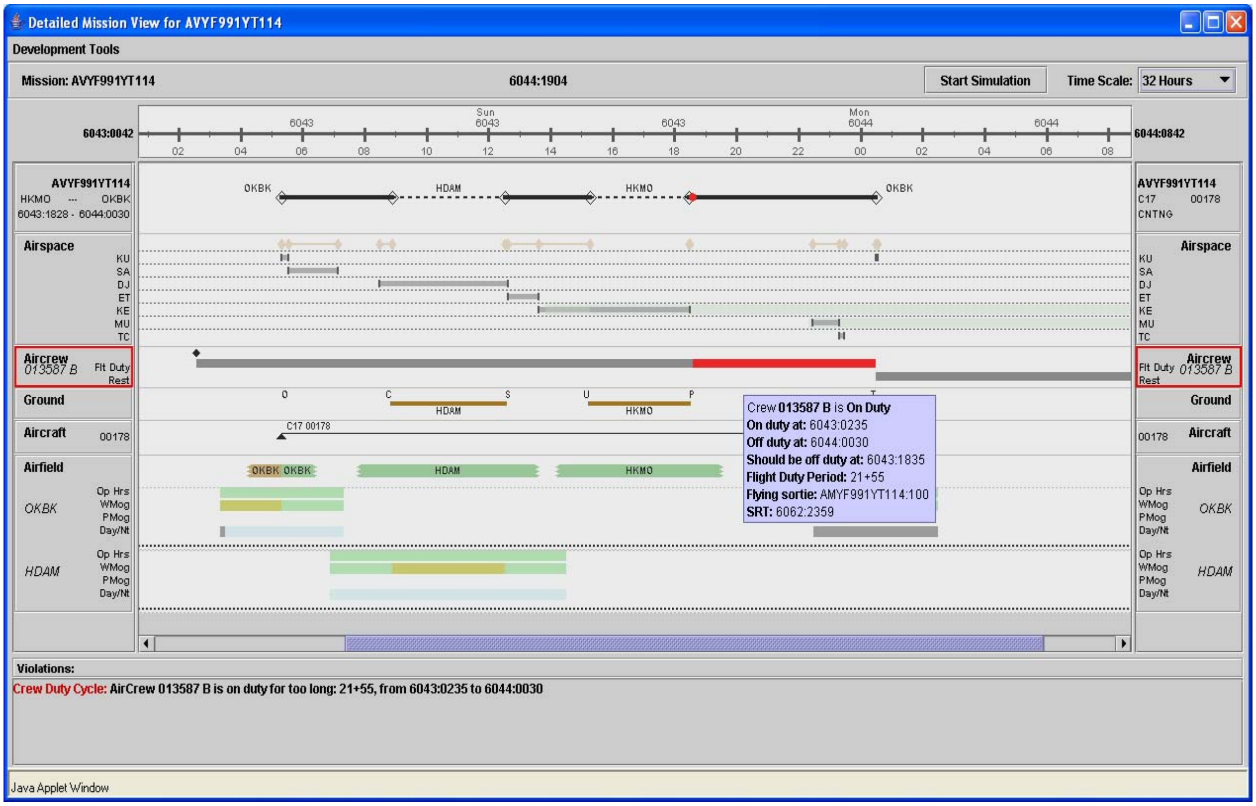
E. Interaction Controls: User input dialogs and rules

i. Mouse-over reveal descriptions displayed as pop-up boxes
ii. Switching between single mission and “Multi-Mission View” (Another WCDP)
iii. Mouse-clicks to reveal/hide contents of each Factor & Resource Cluster.

2. Boundaries of Application: Where HCI solution may be appropriately applied, and not applied

- The timeline is suited to help with identifying itinerary type items of when activities will happen
- Useful when multiple factors and constraints exist that affect the success of an operation
- Does not generally help a user decide on alternative activities

3. **Known Examples**:



Detailed Mission View for AVYF991YT114

Mission: AVYF991YT114

6044:1904

Time Scale: 32 Hours

AVYF991YT114
HKMO --- OKBK
6043:1828 - 6044:0030

AVYF991YT114
C17 00178
CNTNG

Crew **013587 B** is **On Duty**
On duty at: 6043:0235
Off duty at: 6044:0030
Should be off duty at: 6043:1835
Flight Duty Period: 21+55
Flying sortie: AMYF991YT114:100
SRT: 6062:2359

Violations:

Crew Duty Cycle: AirCrew 013587 B is on duty for too long: 21+55, from 6043:0235 to 6044:0030

Figure A4. Screenshot of the spiral 1 timeline tool

Whitaker, R., Scott, R., Roth, E., Militello, L.G., Quill, L.L., Stilson, M.T., Wampler, J.L., Standard, T.W., Thomas-Meyers, G.F., Schmidt, V.A. (2005). *Work-Centered Technology Development (WTD)*. (Final Report, AFRL-HE-WP-TR-2005-0149). Wright-Patterson AFB OH 45433-7604.)

4. **Performance Validation**: Measured impact on target problem

- Air Mobility Mission Replanning (Reference)
  - Impact assessed for replanning of air mobility missions at the USAF Air Mobility Command, Tanker Airlift Control Center (AMC/TACC)
  - Experimental evaluation of scheduling timeline concept vs. baseline HCI in use by AMC/TACC
  - Scheduling timeline concept reduced the number of errors, and reduced the time it took users to recognize impacts of mission changes during execution. Also it improved user's situation awareness of their work and reduced mental workload.

5. Work Content Validation: Methods and measures of HCI correspondence to work as practiced

- ❖ Air Mobility Mission Replanning (Reference)
  - ➢ Cognitive Task Analysis interviews and observations with multiple operator positions within the Tanker Airlift Control Center
  - ➢ Work support evaluation by TACC operators within scenario based test environment

APPENDIX B

Work-Centered HCI Design Pattern: **Task Navigation & Monitoring**

Work Summary

1. Problem Statement: User needs to monitor and control a mix of automated and manual task steps in an operational sequence.

- ➢ Examples: Tomahawk Missile Launch, Shipboard gun control, Shipboard damage control response sequence

- ❖ Each step in an operational sequence is required for safe operations.
- ❖ Steps can be performed in linear sequence or may be done out of sequence where processing logic permits.
- ❖ Steps may be managed by one or more users in same or distributed locations.
- ❖ Challenges can include:
    - ➢ Significant cognitive and procedural burdens involved in evaluating faults that may occur during a procedure and in taking corrective actions.
    - ➢ Maintaining accurate and timely understanding of the operation status.
    - ➢ Potential information overload if monitoring multiple weapon targets and multiple mission flows.
    - ➢ Monitoring and controlling multiple systems and simultaneous missions under reduced crew manning.
    - ➢ Sharing process situation awareness with distributed commanders.

2. Context: The work context normally includes…

- ❖ **Operations Level:** This DP is used to visually depict sequences within *tactical* operations. This DP can be used in a variety of mission domains: Missile or gun activation, launch and monitoring sequence; damage control response sequences; engine propulsion startup or monitoring sequences. The DP may also be applied at an operational or strategic level.

- ❖ **Process Steps**: Processes are dependent on either a prescribed mission plan for offensive warfare, a potential response plan (defensive warfare), or processes are related to resources or physical conditions such as damage control. Examples within Tactical Tomahawk missions: Mission acceptance, mission validation, routing, missile power, etc.

The work context can include…

- ❖ Steps may be performed by one or more users requiring hand-off, reception, and notification of completion across multiple users.
- ❖ A mixture of automated, semi-automated and manual steps.

2. Forces: Conflicting goals or influences include…

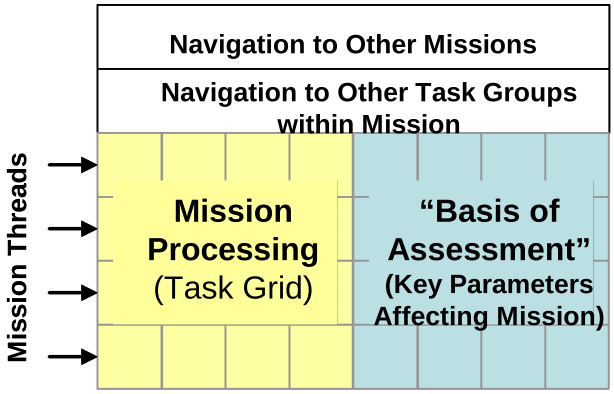
- Monitor one operation process vs. Monitor multiple operation processes.

Solution Summary

1. HCI Description: Task Navigation Controls and Process Monitoring Visualization

A. Intent: Provide a visible task sequence, method of control and HCI navigation with one-button push to navigate to the top-level task information content. Provide monitoring of task processes with user attention management cues to signify immediate task role and response needs across humans and automation. Provide collaborative cues to task hand-off and multi-person approval, authorization and decision-making.

B. Work Field Representation & Schematic: The following figures show a schematic for a generic "step-based" mission. For a discussion of mission sequence types see Osga (12). Figure B1 shows the pattern general representation.



Navigation to Other Missions
Navigation to Other Task Groups within Mission
Mission Threads
Mission Processing (Task Grid)
"Basis of Assessment" (Key Parameters Affecting Mission)

Figure B1. General layout of the Task Navigation & Monitoring DP

i. Central Components

The Task Navigation and Monitoring DP has the form of a "grid" or matrix. The grid contains two main sections – the Mission Processing section (right) and the Basis of Assessment section (left). Each row in the matrix represents a mission thread, e.g. a target or a related collection of targets, a physical location focus (e.g., whatever decomposition makes sense in mission or task). Key features are:

- Time sequence is implied left-to-right but not meant to be specifically serial.
- The pattern information is linked to other information views. Linked items can be highlighted on other information views or can be highlighted when a row is

selected in the grid. For example, a missile route shown on a map view when selected and highlighted also highlights the corresponding mission/target row in this grid.

- ❖ A matrix cell in the "Mission Processing" grid section represents a single step in the process.
- ❖ A matrix cell in the "Basis of Assessment" grid section shows the assessment and explanation for an important mission parameter such as range, location, important physics, safety, approval state, etc.

As shown in Figure B2, each grid square from the Mission Processing section represents a task or step in the mission process. A grid square has the following internal components:

Task Title – a title for the task or mission step
Process Indicator – optional indicator indicating progress of task
Drill-down control – if the task has multiple levels of complexity.

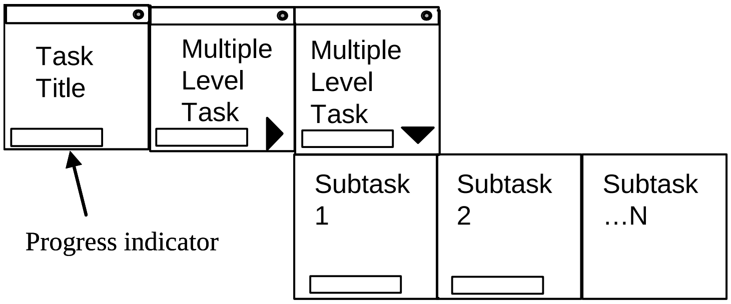


Figure B2. Components of Mission Processing grid squares.

ii. Peripheral Components

Controls are presented peripherally to allow navigation to other task groups within the mission or to other mission domains. These include:

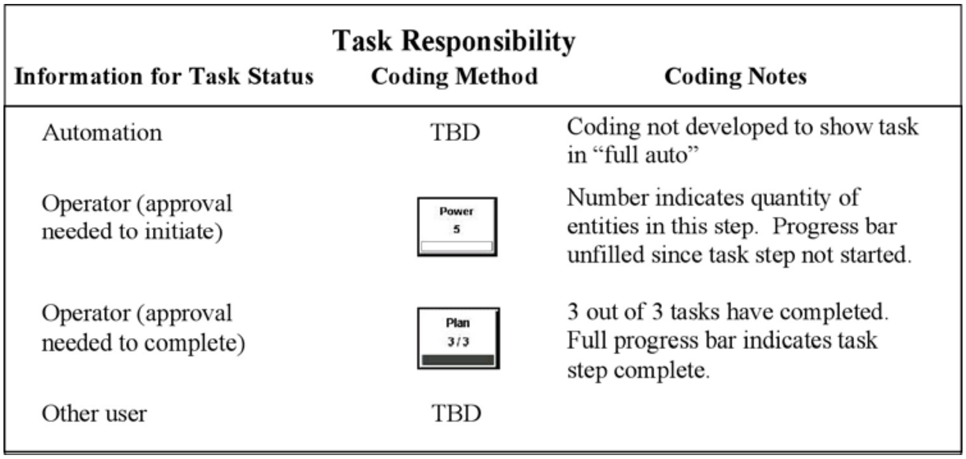
- ➢ **Navigation Tabs (Task Group Tabs)**: Tabs that support navigation to different sets of tasks in a common mission domain.
- ➢ **Navigation Tabs (Mission Type Tabs):** Tabs that support navigation to other mission domains.

C. Contextualized Alerting

As tasks are processed the "business logic" built into the software defines whether a task is ready for user action or if automation is performing the task. Color coding of the grid square is used to indicate the status of the task processing.

i. <u>Central Components</u>
Coding of grid squares is done as follows to indicate status of task processing. The following tables summarize coding used for alerting. The grid cell is colored white if operator action is needed to initiate or approve a completed task.

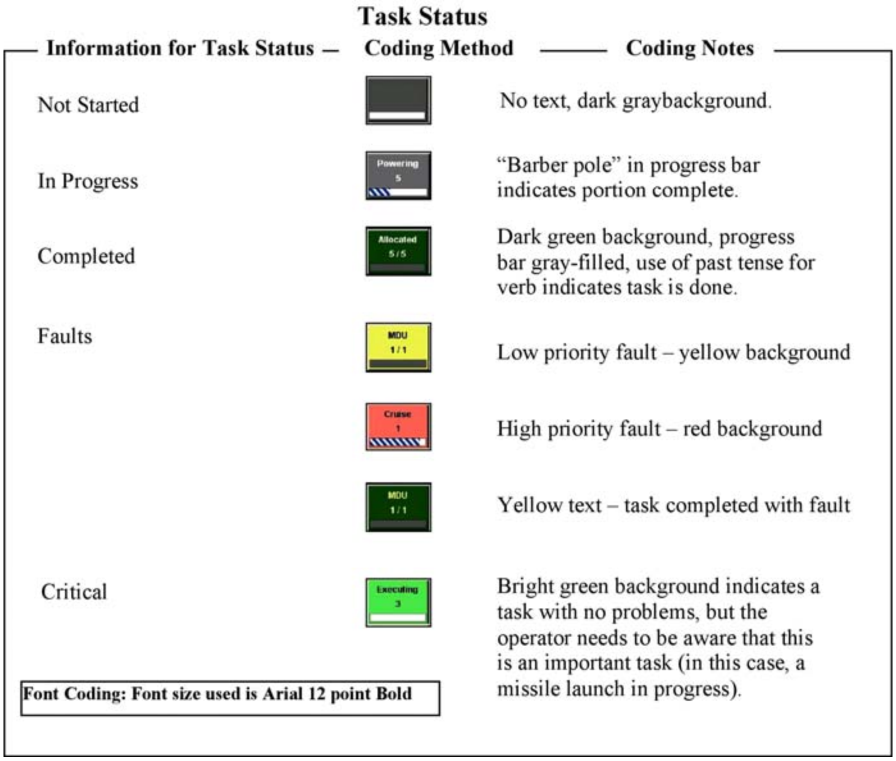
Table B1. Coding of Task Responsibility



	Task Responsibility	
Information for Task Status	Coding Method	Coding Notes
Automation	TBD	Coding not developed to show task in "full auto"
Operator (approval needed to initiate)	Power 5	Number indicates quantity of entities in this step. Progress bar unfilled since task step not started.
Operator (approval needed to complete)	Plan 3 / 3	3 out of 3 tasks have completed. Full progress bar indicates task step complete.
Other user	TBD	

Coding is used to indicate in-progress tasks or faults that may occur during processing. Table B2 summarizes coding used to indicate various task states.

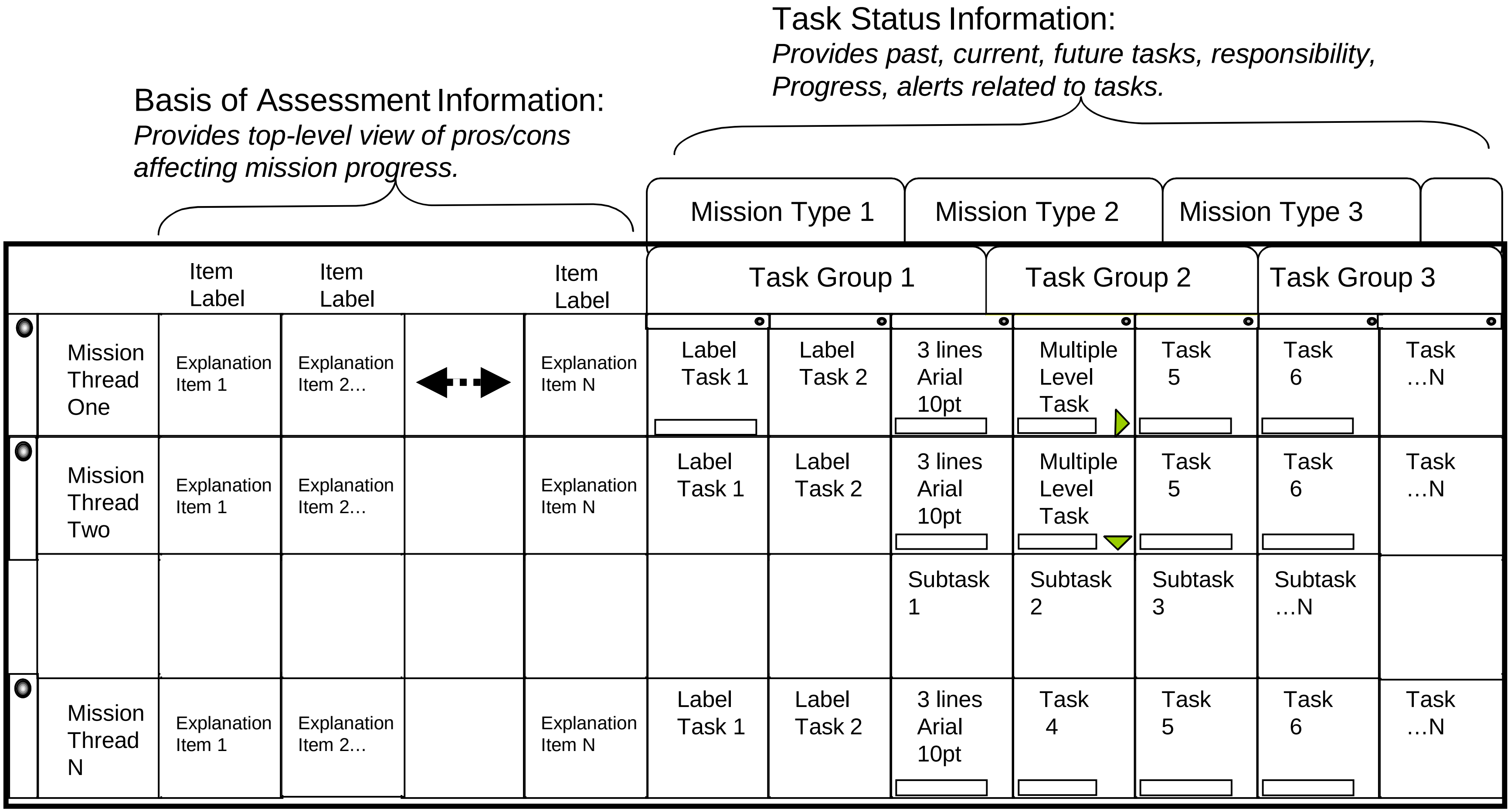
Table B2. Coding of Task Status



	Task Status	
Information for Task Status	Coding Method	Coding Notes
Not Started	⬛	No text, dark graybackground.
In Progress	Powering 5	"Barber pole" in progress bar indicates portion complete.
Completed	Allocated 5 / 5	Dark green background, progress bar gray-filled, use of past tense for verb indicates task is done.
Faults	MDU 1 / 1	Low priority fault – yellow background
	Cruise 1	High priority fault – red background
	MDU 1 / 1	Yellow text – task completed with fault
Critical	Executing 3	Bright green background indicates a task with no problems, but the operator needs to be aware that this is an important task (in this case, a missile launch in progress).

Font Coding: Font size used is Arial 12 point Bold

A template matrix for a "Step-Based" mission is shown in Figure B3.



Task Status Information:
Provides past, current, future tasks, responsibility, Progress, alerts related to tasks.

Basis of Assessment Information:
Provides top-level view of pros/cons affecting mission progress.

						Mission Type 1		Mission Type 2		Mission Type 3		
		Item Label	Item Label		Item Label	Task Group 1			Task Group 2		Task Group 3	
●	Mission Thread One	Explanation Item 1	Explanation Item 2…	◀···▶	Explanation Item N	Label Task 1	Label Task 2	3 lines Arial 10pt	Multiple Level Task ▶	Task 5	Task 6	Task …N
●	Mission Thread Two	Explanation Item 1	Explanation Item 2…		Explanation Item N	Label Task 1	Label Task 2	3 lines Arial 10pt	Multiple Level Task ▼	Task 5	Task 6	Task …N
								Subtask 1	Subtask 2	Subtask 3	Subtask …N	
●	Mission Thread N	Explanation Item 1	Explanation Item 2…		Explanation Item N	Label Task 1	Label Task 2	3 lines Arial 10pt	Task 4	Task 5	Task 6	Task …N

Figure B3. Pattern for Multiple-Threads in a Step-Based Mission

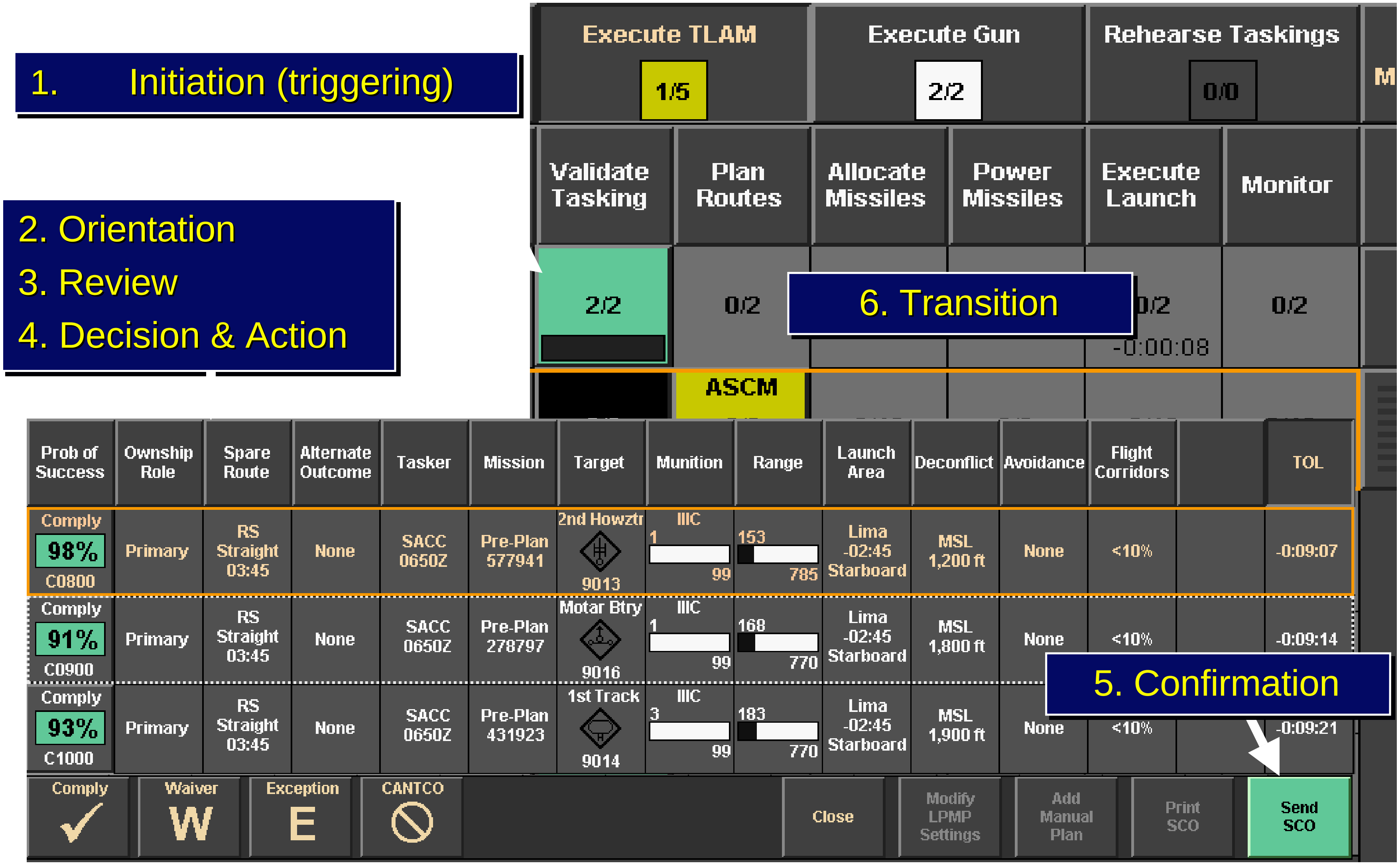
D. Interaction Controls Description: User input dialogs and rules

- Mouse-over can reveal descriptions displayed as pop-up boxes
- A single click displays the task information set including controls and decision support information to support the task.
- A double-click can be assigned to other navigation.

2. Boundaries of Application: Where the HCI solution may be appropriately applied, and not applied

- The grid is applied to discrete event sequences where variability in the sequence is relatively low and stable.
- Useful when multiple factors and constraints exist that affect the success of an operation
- Less useful for non-linear and informal task sequences – consider other patterns such as a checklist DP
- For larger information sets with many parameters in the Basis of Assessment (B of A) an alternate format is used: The Task Status Information and B of A are separated *vertically*, with the B of A expanding underneath the Task Status Information.

3. Known Examples: Tomahawk Launch Sequence with cognitive steps 1-6 shown in Figure B4.



1. Initiation (triggering)

2. Orientation
3. Review
4. Decision & Action

5. Confirmation

6. Transition

Prob of Success	Ownship Role	Spare Route	Alternate Outcome	Tasker	Mission	Target	Munition	Range	Launch Area	Deconflict	Avoidance	Flight Corridors		TOL
Comply 98% C0800	Primary	RS Straight 03:45	None	SACC 0650Z	Pre-Plan 577941	2nd Howztr 9013	IIIC 1 99	153 785	Lima -02:45 Starboard	MSL 1,200 ft	None	<10%		-0:09:07
Comply 91% C0900	Primary	RS Straight 03:45	None	SACC 0650Z	Pre-Plan 278797	Motar Btry 9016	IIIC 1 99	168 770	Lima -02:45 Starboard	MSL 1,800 ft	None	<10%		-0:09:14
Comply 93% C1000	Primary	RS Straight 03:45	None	SACC 0650Z	Pre-Plan 431923	1st Track 9014	IIIC 3 99	183 770	Lima -02:45 Starboard	MSL 1,900 ft	None	<10%		-0:09:21

Figure B4. Tomahawk Weapon System Launch Sequence with cognitive steps 1-6 shown.

4. Performance Validation: The following list includes studies conducted using same or similar task grid designs.

- Tactical Tomahawk Mission Control.
  - Cognitive workload reduced as measured by a primary/secondary task paradigm tested with naval operators (B1)
  - Usability testing results (B2, B3, B4). Easily trained and understood by novice and advanced operators.
  - Workload reduced within high-tempo, multiple target scenario with one operator performing all the work normally performed by four operators (B5)

(B1) Brooks, B. (2003). Empirically Validating System Design: The Results, Conclusions, & Implications of TTWCS Summer 2003 Usability Testing. Unpublished Presentation. Lockheed-Martin Mission Data Systems, Valley Forge, PA: Author.

(B2) Borja, A., Kellmeyer, D., & Edwards, B. (2003). Task Manager for TTWCS v5 Usability Evaluation Report. (unpublished Tech Report, February) Space & Naval Warfare Systems Center. San Diego, CA: Author.

(B3) Borja, A., Kellmeyer, D., and Chang, M. (2003). LACS RPT Version 12.2 Usability Evaluation Report. (unpublished Tech Report, August) Space & Naval Warfare Systems Center. San Diego, CA: Author.

(B4) Kellmeyer, D. & Borja, A. (2003). LACS Rapid Prototype Version 10.2 Usability Evaluation Report.. (unpublished Tech Report, April) Space & Naval Warfare Systems Center. San Diego, CA: Author.

(B5) Pharmer, J., Cropper, K., McKneely, J., Williams, E. (2004) Tactical Tomahawk Weapon Control System v6 Land Attack Combat System Prototype Human-Computer Interface: Test Report for FY04 Fleet Operability Test. Technical Report 3184 Space & Naval Warfare Systems Center San Diego. July.

(B6) Osga, G, Linville, M., Kellmeyer, D. Griffith, C., Williams, E., Feher, B., Adams, R., Lulue, D., Burt Edwards (2006) Advanced interface Display (AID) Guidelines and Lessons Learned. Technical Report (in progress) SPAWAR Systems Center San Diego.

5. Work Content Validation: Methods and measures of HCI correspondence to work as practiced

- Tactical Tomahawk
  - The tasks selected for the matrix matched the current legacy system v4.0 Tomahawk control software.
  - Multiple usability tests conducted with Tomahawk operators and training instructors.