WORK-CENTERED DESIGN: A COGNITIVE ENGINEERING APPROACH TO SYSTEM DESIGN

Robert G. Eggleston
Air Force Research Laboratory
Human Effectiveness Directorate
Wright-Patterson, AFB, OH

This paper provides an overview of an emerging cognitive engineering framework that can be used to guide engineering analysis and design decision-making. As the name implies, the Work-Centered Design (WCD) framework considers the design problem from the perspective of an analysis of work. It pays special attention to an analysis of aiding requirements, including issues about direct automation and form-based approaches for aiding. The paper provides a brief description of the WCD framework, identifies novel features and illustrates selected concepts with examples.

INTRODUCTION

The tools and systems that workers use in work today continue to grow in both capability and complexity. As a result, we must continue to improve our ability to provide better user interfaces to these tools and to the critical characteristics of the product work itself. This implies that we need to continuously examine our design tools, methods, techniques, and processes to determine how they might be improved to aid the development of the user interface, in particular, and human-centered technologies, in general, to better assist users during work.

My colleagues and I are in the process of developing information technology (IT) applications called work-centered support systems (WCSS) (Eggleston et al., 2000; Young et al., 2000). A WCSS is a user interface technology conceived from the perspective that the ultimate goal of the user interface itself is to aid the user in accomplishing work goals. It can be regarded as a form of an assistant or associate that aids with work. This form of aiding technology is unique in that the entire interface is treated as the aid, and machine automation is blended with context factors and representational form factors to achieve an immersive information-interaction workspace. In the course of developing WCSS applications, some new ideas about the design and evaluation of human-centered technologies have emerged. The purpose of this paper is to briefly describe this emerging design framework, which we call Work-Centered Design (WCD). The paper provides a brief description of the WCD framework, discusses key design issues and analysis needs, and illustrates important analysis concepts with examples.

WORK-CENTERED DESIGN

The WCD framework focuses on the construct of work. Work represents the interaction between workers and tools, machines, and systems used by a person in the context of meeting the goals and product requirements of a work problem. Work may be defined in broad terms as the roles

---

Figure 1. Overview of the Work-Centered Design (WCD) framework. Analysis and design tools, methods, and techniques are identified for each process in the framework.
and responsibilities of a person (Vicente, 1999). To meet responsibilities, a worker must be prepared to handle planned tasks in a dynamic context. This means that work involves addressing unexpected situations and disruptions, and possibly the need to modify how standard tasks are performed in light of the prevailing situation. Therefore, work is different from simply accomplishing a preplanned set of steps in a prescribed way.

In the WDC framework, work is conceived of as a dynamic process that encompasses multiple facets, including problem solving/decision making, collaboration, product development, and work management. Each of these facets identifies a different referent that must be addressed in the design process. Current approaches to human-centered design do not always consider each of these perspectives.

The WDC framework is consistent with, and in many ways, overlaps other methods to design that are known in broad terms as the cognitive engineering approach to human-centered design. This approach pays particular attention to the cognitive work that must be performed in meeting user goals and action requirements. The WCD is unique in that it emphasizes a coordination of contexts as a central factor that must be designed to better insure the interface system aids the worker in completing their responsibilities.

An overview of the WCD framework is presented in Figure 1. In brief, the Work Knowledge Capture process collects raw data to be used throughout the design process. The Work-Centered Requirements Analysis process defines intrinsic properties of work and thus sets the foundation for work aiding. The Work Aiding Design process further defines work in ways that address flexible and adaptive aiding in context. Finally, the Work-Oriented Evaluation process provides a method of assessment that is expected to provide a comprehensive evaluation and also to facilitate product transition. The four design and development functions or processes follow a logical order of application, as suggested by the figure, but in actual practice, activities in different processes may occur in parallel, and, of course, multiple iterations through the processes are normal.

Work Knowledge Capture

Once the customer's needs, goals, and requirements have been expressed and are understood, initial activities focus on capturing knowledge about the organization as a work system. This is a discovery process. It has several aims:

- To gain familiarization and deeper understanding of the customer's stated needs, goals, and requirements
- To ground this understanding in terms of work within the context of the organization; and
- To gain a solid understanding of work as actually practiced by individual workers in context.

Several methods are used to capture information that affords development of this knowledge. While work is inextricably bound to an agent (human or machine automation), some of the knowledge capture methods telescope out to the broader context in which the specific work of interest is contained. This includes acquiring information about the business process model for the organization and job descriptions for the interacting job position(s) of interest. This information tends to be available in abstract and analytical form. Other methods attempt to focus in on concrete work practices, such as the use of observational studies, question probing techniques, and the detection and analysis of locally produced work aids or artifacts. Because work involves flexible and adaptive behavior, we attempt to capture information about solving hard problems, what interruptions occur and how they are handled, and the processes for directly achieving work goals. The goal is to build up an understanding of the richness and complexity of work and the work context.

Work-Centered Requirements Analysis

The captured knowledge is further analyzed in order to identify design requirements that stem from work itself. The basic aim of these analyses is to increase understanding about work in general and to ferret out intrinsic properties of work in which the customer's requirements are embedded. In other words, we attempt to develop a formal model of work that attempts to capture its intrinsic nature; and hence is technology independent.

We have found it to be useful to follow the practice of separating out aspects of work into event-independent domain properties, event-dependent process or activity properties, and agent strategies, as advocated by Rasmussen and his colleagues (Rasmussen, 1986; Rasmussen et al, 1994; Vicente, 1999).

The goals, values, and basic functions derived from a work domain analysis identify some of the intrinsic properties of work for the position(s) involved. Additional properties of work are derived from a process analysis. The goal is to uncover the natural and logical partitioning and sequencing of work activities that are inherent in the domain.

Consider the work of a Weather Forecast Professional (WFP) in the context of a military airlift organization. As the name implies, this worker engages in weather forecasting activities. Because the airlift organization operates world wide, forecasting must cover this space over times relevant to flight route planning and replanning during execution. Natural partitions for this work occur. Meteorological phenomena are characterized at different scales of space-time resolution and may logically be addressed from different geographical vantages. For example, a synoptic scale of analysis is used to understand weather in terms of atmospheric disturbances in the form of fronts, cyclonic and anticyclonic activity, etc. A smaller space and time scale (mesoscale) is used to understand weather phenomena in terms of surface and upper wind patterns and convection and thunderstorm outlook. Further, specific weather phenomena elements (e.g., clouds, visibility, precipitation, etc.) germane to aircraft flight activities (take-off, landing, and enroute) represent an additional partition by region and by specific route plan. Given the rate at which weather phenomena persist and change, these are all natural
space-time partitions to impact weather forecasting work. While these attributes of weather forecasting work depend on the dynamics of weather that form patterns known as weather events, these properties of work are independent of any tools and technology used to aid the forecaster in work.

Based on an understanding of the intrinsic nature of specific work in its context, different types of design requirements are generated. These include functional, informational, decision making/problem solving, and situation awareness requirements.

However, a formal model of work must represent other aspects of work as well. A complete work analysis requires an examination of factors that arise as a function of the dynamics of the work process itself. These factors may be regarded as metawork that are incidental to the main goal of work, but that arise from work dynamics. We use the terms Work Management and Collaboration as labels for two major aspects of metawork. Our complete formal model of work includes four aspects: Work Management, Collaboration, Problem Solving/Decision Making, and Product Development. Problem solving/decision making (DM for short) spans intrinsic work and metawork. Product development refers to the output products of work, whether they are tangible or intangible (Eggleston et al., 2000).

**Work Aiding Design**

In some sense, work requirements themselves constitute designing. The requirements express goals to be achieved or constraints to be satisfied and thus, they limit the available design space. Thus, even thought they do not specify the actual from of the to-be-designed artifact, they influence its form and in this sense are part of the design. Additional work-oriented specifications about the design form itself are needed to provide necessary guidance to the developer who expresses the design in the final medium. These specifications may be represented by the output of further analysis that is motivated from the perspective of user aiding or they may be best represented by renderings of design concepts. Rendering or illustrations are particularly important when the aiding specification is based on the form of expression itself.

The aim of the Work Aiding Design analysis is to identify user-aiding requirements that are useful and that have properties that allow them to be applied to work by the worker in a flexible and adaptable manner, given the dynamics of the work situation. This is an area of design analysis that we believe has been neglected in the past. Current cognitive engineering and other design approaches have under-specified the form requirements of the design and left them for the final implementer to determine. As an initial effort to provide more precise form specification from a work-centered perspective, we have developed some new analysis methods.

One of these new methods considers the language to be used in the to-be-designed artifact. In complex work domains, it is common to find the same term given different meanings, and conventional terms that have lost contact with their original referent but have become part of the fabric of the domain. It is important to capture and analyze the terms and their relations (i.e., the ontology) of the work domain and to consider how they contribute to aiding work. One goal is to understand what contextualized meanings for a term need to be retained in the support system representation to provide proper aiding. Another goal is to know what terms are now, de facto, part of the fabric of the domain and thus must be preserved in the representation, even though on traditional analytic grounds a different term may be more desirable. The ontology analysis is especially important if system automation includes the ability to form meaning during the dynamics of work and the ability to act on this emergent semantic basis.

A second area where more analysis is needed to advance the aiding specification is in terms of what and how (at a category levels) it should be expressed in the support system design. We suggest that aiding can be provided in both direct and indirect forms (Eggleston et al., 2000). Direct aiding is provided by machine automation. (We use software agents.) The aiding may focus on information fusion, problem analysis, or information presentation. Or it may focus on assisting, augmenting, or substituting machine manipulation and control in lieu of human performance of these actions.

The form of the *aiding relationship* between the human and the machine must also be designed. These are difficult problems and they are largely being addressed in an ad hoc manner today.

We are currently attempting to determine two aspects of aiding. First, what are viable direct and indirect aiding methods for a specific issue or problem? Indirect aiding uses representation and visualization to guide worker understanding of the problem in the surrounding context. The goal is to provide a representation of the work field that shows the problem and also shows possible solution pathways that are open. It presents aiding in terms of a field representation of the work domain. Second, when direct aiding solutions are viable or perhaps preferable, our analysis attempts to discover ways to express the aiding in the ontology of the work itself. That is, aiding is embedded in the structure of work in a natural manner. For example, we have produced a prototype work support system to aid weather forecasters in predicting the impact of weather phenomena on aircraft routes and missions. If weather characteristics in a region are estimated to have a potential to cause a problem for the current planned route, the forecasters can launch a software agent to aid her in keeping tabs on the weather in the area of interest. The automation is expressed in terms of a watch box that is drawn by the forecaster on a map display by pointing and clicking a mouse, and by specifying parameters values to be used to set the conditions for an alert. The agent is visible in terms of weather characteristic and weather work, and not as an agent, per se. It is expressed in the work ontology (Scott et al., in press).

We believe it is important to understand work from the perspective of the worker in context, not just in more general intrinsic work terms. Context is derived from 1) the nature of work, 2) disturbances and interruptions that arise as work unfolds, and 3) by the state and viewpoint of the worker. In order to better understand what aiding a worker might find useful, we need to know how the problem appears to the worker. To address this issue, we have proposed an analysis technique called Problem Casting. It takes an unscheduled
event in the environment that influences the work product as a problem to be addressed or solved by the worker. Given this problem, it asks: what are possible ways the worker may think about this problem? The goal is to understand 1) what support may be useful, and 2) what navigation may be useful to provide ready access to an appropriate work context and information sources. This addresses the issue of flexible work aiding.

For example, in airlift mission planning work, a common problem occurs as new missions are planned for execution during the same time period as previously planned missions. One problem of this type is called a MOG or maximum-on-ground problem. Parking MOG is the problem where there is not enough parking slots available at an airport at a given point in time for a given aircraft size class. Flight schedules for the conflicting aircraft must be adjusted to avoid delays. There are at least three different first-person perspective ways a mission planner may think about this problem in the prevailing work context when it arises. One is to decide to suspend current work and directly handle the MOG problem. Another is to see if the owner of one of the other missions may be able to handle the problem. And a third view might be to notice that the problem is not as urgent as other work and thus can be scheduled for attention later. Each of these views expresses a different way the problem can be cast. Casting analysis attempts to determine the set of reasonable and most likely problem castings. Then further analysis is performed to establish a suitable specification that takes these aspects of work into consideration.

Illustrative rendering are an important tool in design aiding analysis. Some aspects of aiding are difficult to analyze and to express as design specifications without the use of a drawing. This is true for form-based requirements and to address issues important to contextualizing aiding in the support system.

Currently, notional display format layouts, user-case, and scenario-based storyboards are methods used to analyze and express design requirements. These techniques can capture context effects, and we use them, but they are not geared specifically toward aiding analysis.

We believe a focused analysis on context patterns is important. These patterns have much to do with the flexibility and adaptability afforded by a support system design. Thus, it is important that they be suitably analyzed and specified.

We have not yet developed formal methods in this regard, but we have provided the problem-vantage-frame principle as a general guide to designers that in part addresses the contextualization issue (Eggleston and Whitaker, 2002). According to this principle, the product goal acts as a contextual frame, the current problem reflects a different contextual frame, and the way the worker’s “felt demands” are interpreted by the worker establishes her vantage on the problem as a third contextual frame. Aiding needs to achieve a coordination across these co-occurring contexts. This means both target work and metawork aspects of the context must be considered. Proper coordination of the contexts and their expressions support fluid, flexible, and adaptable work patterns.

To illustrate the point, consider Flight Management (FM) work. In this type of work, a FM plans all of the details of a flight route for an aircraft that has a specific mission. Once the mission is initiated, the FM also tracks and interacts with the aircraft. She responds to requests and attempts to anticipate problems and develop corrective solutions before a problem is made manifest. A FM may be responsible for on the order of 20 missions per shift.

Several contexts and scales of work come into play for the FM. First, the FM must manage her work schedule. Second, because of the intrinsic nature of the planning work, information availability for selected planning variables is spread out over time. As a result, the FM typically must be working on several plans simultaneously. Thus, use of the same set of tools must be switched between each mission context, and the planning work, in general, must switch back in forth across different missions. At the same time, the FM must also detect emerging problems and disturbances across all of the missions and handle interruptions from other planners, senior management, and the flight crews. In short, patterns of work span many different contexts, and the worker must be able to 1) rapidly become grounded in an seset of contexts, 2) rapidly and intelligently switch between context sets, and 3) attempt to achieve fluid work over the entire domain.

Figure 2 presents a rendering of a part of a work-centered support system that addresses work aiding for the FM. A list of missions or sorts that define the work problem for a work shift is provided on the left. Hence its primary frame is the metawork context of work management. However, it includes within it information elements related to the target work domain as well. It provides a means for the FM to understand the work situation globally, with embedded information that is useful for local understanding. The larger panel on the right is a work frame for a single mission that must be planned. A separate frame is spawned for each mission. A frame includes a work area where different applications may be launched and used to accomplish specific work activities. The states of these tools and information are preserved for each mission in this manner. On the right side of this frame is an interactive “check list” that addresses metawork inside a specific mission context. It indicates at a glance what planning work has been completed and what remains open. This is important for rapid re-grounding as the FM moves across several missions and responds to interruptions and emergent problems. There are more aspects to the aiding being illustrated by this dawning, but suffice it to say that the drawing, along with descriptive information, serves as a means to communicate a design specification that addresses what context coordination and formed-based aiding needs to be included in the design. By specifying the aiding in terms of a coordinated pattern of context fields, we identify the “cognitive sweep” of the work that must be included in an aiding solution. At the same time, degrees of freedom are open to the worker for efficient, flexible, and adaptive work. Degrees of freedom are also open to the implementer in terms of how to generate the context patterns and express them in the final medium.
WORK-ORIENTED EVALUATION

We advocate an evaluation method that concurrently assesses usability, usefulness, and impact of each design prototype. This is different from most current practice. The aim is to insure that each interactive has work value in its own right thus provides an opportunity for rapid transition. To achieve this goal, we implement prototypes in terms of work threads and evaluate the prototypes in terms if usability, usefulness, and impact, where impact is specified at three levels: bench, unit, and senior management (Eggleston et al., 2003). Different measurement techniques may be used within this work-orientated framework. We have employed typical feature-use exercises to address usability, an interactive work scenario method to assess usefulness, and a paper and pencil survey method to gain additional usefulness measures and impact or value measures that span the bench, unit, and senior level domains of interest in product evaluation. A more detailed discussion of this evaluation framework is available in Eggleston et al., 2003).

CLOSING COMMENTS

The cognitive complexity of work continues to grow for an ever-increasing segment of the work force. Information technology can be part of the solution, but it is also clearly part of the problem. Information overload is commonplace, and growing. There is no doubt that we need better work-centered designs. To produce them we also need improvements in our design toolkit. I have introduced the Work-Centered Design framework in this paper. It should at least make the goals of the framework clear and provide some insights into how we are trying to meet these goals. We think it is important to treat the user interface as a support or aiding system in its own right; that design and analysis must span target work and metawork; that patterns of contexts must be analyzed and specified; and that a comprehensive range of assessments need to be completed for each prototype cycle. Some of these areas of analysis have received little or no formal attention in the past. I believe they are critical if we are going to be able to systematically and reliably develop designs that enable the worker to better cope with complexity and achieve a desirable level of performance.

REFERENCES


