Proceedings of the Human Factors and Ergonomics Society Annual Meeting
http://pro.sagepub.com/

Toward an Integration of Task- and Work Domain Analysis Techniques for Human-Computer Interface Design

Christopher A. Miller and Kim J. Vicente

Proceedings of the Human Factors and Ergonomics Society Annual Meeting 1998 42: 336
DOI: 10.1177/154193129804200333

TOWARD AN INTEGRATION OF TASK- AND WORK DOMAIN ANALYSIS TECHNIQUES FOR HUMAN-COMPUTER INTERFACE DESIGN

Christopher A. Miller
User Centered Design Technology Group
Honeywell Technology Center
Minneapolis, Minnesota

Kim J. Vicente
Cognitive Engineering Laboratory
Department of Mechanical and Industrial Engineering
University of Toronto
Toronto, Ontario

Work analysis techniques are critical, longstanding methods for designers to obtain the knowledge required for good interface design. The majority of current techniques fall into two types: task-centered and system- or work domain-centered. These approaches have different and largely complimentary strengths and weaknesses, but they focus on different aspects of the design problem and a unification is required for completeness. We discuss and compare the characteristics of both approaches. Then we present results and lessons learned from an attempt to integrate two characteristic analysis techniques in analyzing interface requirements for a simple feedwater system: Rasmussen's (1985) Abstraction Decomposition Space (also known as the Abstraction Hierarchy) and Sewell and Geddes (1994) Plan-Goal Graph.

INTRODUCTION

Human factors engineering begins with work analysis. This is true both of the field's history, (e.g., Tayloristic time and motion studies), and of most textbooks' recommendations for proceeding with interface design (Booth, 1988; Norman, 1989; Wickens, 1992). As Woods (1991) puts it "...the design of displays/interfaces/aids is shaping or processing display surfaces . . . so as to exhibit information for a domain practitioner" but "information is a relation between the data, the world the data refers to, and the observer's expectations, intentions and interests" (Woods, 1986). Thus, to know how to shape a display surface, the designer must know what data are pertinent to the observer's expectations, intentions and interests, and in what order and relations. This knowledge is provided by work analysis methods.

We observe that most work analysis techniques can be divided into two types based on their primary focus. Task analysis (Kirwan & Ainsworth, 1992; Diaper, 1989) describes the actions that an actor can or should take to accomplish goals. Work domain analysis techniques (Rasmussen, Pejtersen & Goodstein, 1994; Rasmussen, 1985), by contrast, examine the functional structure of the domain (specifically, the plant or system) in or on which work must be done.

Each approach has strengths and weaknesses, but ultimately they reflect different perspectives on (and different avenues to) the knowledge needed for human-centered system design. We believe that a unification of the alternate approaches is required to obtain the complete set of knowledge needed for good interface design. Without such unification, designers and analysts may still be able to obtain sufficient knowledge for good interface design, but only in an indirect and implicit fashion. As systems and their interfaces become more complex, this approach will be both inefficient and error prone. We will discuss both task and work domain analysis approaches in separate sections below and then present initial results from research focused on integrating them to serve interface design for a petrochemical processing application.

TASK-BASED ANALYSIS AND DESIGN METHODS

Task analysis (TA) techniques have a long and productive history in human factors. Kirwan and Ainsworth (1992), in their comprehensive work on the vast variety of TA methods, define TA "... as the study of what an operator... is required to do, in terms of actions and/or cognitive processes to achieve a system goal." Thus, TA methods are explicitly about the actions that an actor can or should take to achieve a goal. That those actions are taken with or on a system shows that an analysis of the system should be included to provide a robust understanding of the work to be done. The focus of TA is the action, however, not the work domain. Knowledge about tasks captured in analysis typically includes either hierarchical, means-ends relationships (how subtasks may be composed to accomplish higher level tasks) or sequential relationships (how tasks must be performed temporally in order to be successful), or both. Sources of information for TAs are typically user interviews, though observation, experimentation and training or procedural manuals may also be used (Diaper, 1989). Where these sources are absent, and in those circumstances where task knowledge breaks down (e.g., unanticipated situations), TA will be impossible, or worse, misleading. When these sources do exist reliably, however, failure to incorporate them into design will result in inefficiencies or errors in training and operations.

Information needs (both input and output) are typically deduced for the tasks and these, combined with the task relationship information described above, can serve as the basis for prioritizing, clustering, filtering, or sequencing information presentation elements in an interface design. Task-linked information requirements serve as a particularly powerful basis for constructing "context" sensitive (actually, user intent, goal or procedure) interfaces (Funk and Miller, 1997) since they can dynamically filter information on the basis of the current user information needs (Rouse, Geddes, and Curry, 1988; Miller, Funk and Hannen, 1997).

WORK DOMAIN ANALYSIS AND DESIGN METHODS

Work domain analysis (WDA) techniques are more recent additions to the repertoire of interface design tools. Most current work in this area derives from Rasmussen's (1985) abstraction-decomposition space (ADS)—commonly, if somewhat incorrectly, referred to as the 'Abstraction Hierarchy'. An ADS is a two-dimensional modeling tool that can be used to conduct a WDA in complex sociotechnical systems. Rasmussen's approach, shares the Gibsonian (Gibson & Crooks, 1938) emphasis on the importance of the "field" or ecology in which an actor behaves for determining or "constraining" the set of actions which are necessary or appropriate. There is a growing amount of empirical support showing that interfaces based on such WDAs can lead to better performance than traditional interface approaches, especially under abnormal situations (Vicente, 1996).

Ecological interface design (Vicente & Rasmussen, 1992) is a theoretical framework that has been developed for constructing interfaces on the basis of an ADS. The ADS provides a comprehensive analysis of the means-ends and part-whole relationships in the functional structure of the process being controlled. It is important to note, however, that while some TA approaches represent means-ends relationships, these are 'action' means-ends (i.e., what actions need to be performed in order to achieve ends at a higher level). By contrast, an ADS represents 'structural' means-ends relationships (i.e., what structural states of the system are required in order to achieve higher level ends).

Ecological interface design uses knowledge of constraints in the process to derive, and then directly display, components available to effect process state changes as well as the difference between the expected and actual behavior of the plant. That the operator must take actions on that process again points to the fact that knowledge about actions should be included in the interface design, but the primary focus of WDA techniques is the plant or system, not operator actions.

WDA relies on a detailed knowledge of the plant and its interactions with the environment—and on the rules, equations or models governing these interactions. When these sources are inadequate, the analysis will be correspondingly inadequate—but this situation is less common than might be expected. The greatest threat to the safety of process control systems is events that are not familiar to operators and that have not been anticipated by designers (Vicente & Rasmussen, 1992). Under these challenging circumstances, the operator's role is one of adaptive problem solver. Because the event has not been anticipated by system designers, the available procedures, experience, and automated aids are not directly applicable. The one thing that does remain unchanged, however, is the functional structure of the plant and the principles that govern its interactions with the environment. Further, it is precisely within these constraints that the operator must improvise a solution.

COMPARISON OF THE TECHNIQUES

Task-based models are like directions for navigation: they identify the actions that human operators should take for particular situations; system-based models are more like maps because they emphasize the overall structure of the plant, independent of any particular situation. Task models are efficient because they identify the information and prioritize it for pre-defined classes of situations, whereas system models are more robust because they identify the functional relationships that are potentially relevant for all situations.

Table 1 compares TA and WDA. TAs (and interfaces designed from them) are efficient because they identify what needs to be done, and perhaps how. But as a result of this economy, TAs do not provide the support required to adapt to unanticipated events. TAs are narrow in their generality because they are only applicable to the tasks that have been identified up front, and generally, only to specific ways of doing those task. In task-sensitive interfaces, efficiency is accomplished by suppressing information not pertinent to specific tasks at hand, but this may risk loss of accurate, overall knowledge of process state. While context-sensitivity can be accomplished by adapting the interface to specific work domain states, this frequently presupposes an implicit task-orientation and may undercut the comprehensiveness of information availability described above. Finally, again due to their narrow, brittle, procedural orientation, TAs are also limited in their ability to support recovery from errors.

WDAs (and interfaces derived from them) have a complementary set of strengths and weaknesses. Their pri-

Table 1. Relative advantages and disadvantages of TA and WDA forms of work analysis (and, by extension, to interfaces designed from information obtained via these analytic techniques).

	TASK	WORK DOMAIN
Mental economy	efficient	effortful
Ability to adapt to unforeseen contingencies	brittle	flexible
Scope of applicability	narrow	broad
Ability to recover from errors	limited	unlimited

mary disadvantage is that they do not tell workers what to do or support them specifically in what they are currently doing. As a result, WDAs put greater demands on workers and may lose efficiency by failing to support specific methods that are known to work in specific conditions. Yet WDAs are generally flexible because they provide workers with the information they need to generate an appropriate response, on-line in real-time, to events which have not been anticipated by system designers. Moreover, WDAs also have a broader scope of applicability. Because they show what the system is capable of doing they provide workers with the discretion to meet the demands of the job in a variety of ways that suit their preferences or the particular needs of the moment. Finally, for the reasons already discussed, WDAs also provide workers with the support they need to recover from errors.

Because they have complementary strengths and weaknesses, it would be useful to include both WDA and TA techniques in a single, integrated framework for work analysis and interface design. In this way, TA can provide efficiency to deal with predictable tasks, and WDA can provide the breadth and generality to cope with unanticipated events. In fact, comprehensive approaches to Cognitive Work Analysis (Vicente, in press) have realized that both tasks and work domain structure, as well as social organization and human competencies, are necessary to fully understand the work environment. Our work is an attempt to use more powerful task representation techniques, and to integrate them more closely with work domain representations, than has typically been attempted in the past.

TOWARD A UNIFIED ANALYSIS TECHNIQUE

Together with NOVA Chemicals Ltd., we have begun research toward an analysis technique that will support modeling both tasks and work domain constraints and the relationships between them. As a test case, we have begun by applying a modified form of Sewell and Geddes (1990) Plan/Goal Graph (PGG), a TA technique which emphasizes means-ends relationships between tasks and the goals they accomplish, to Vicente's DURESS II simulation (Vicente, 1996). DURESS II is a thermal-hydraulic process system simulation, which has been extensively modeled using ADS techniques (Bisantz & Vicente, 1994). Although there is substantial local expertise in operating DURESS II, TA modeling techniques have not been applied and task-based interfaces have not been designed for it. Thus, it serves as a good domain for developing and testing initial theories and techniques before moving into the more realistic petrochemical domains that NOVA offers.

We began by attempting to construct a task model that paralleled and integrated the work domain information contained in the ADS model of DURESS II. As a TA technique, PGG is exceptionally good at representing the means-ends relationships between tasks and the goals they accomplish (albeit, in terms of action relationships, not in terms of the structural work domain relationships). However, it is less good than some other TA techniques at explicitly representing known sequential ordering of tasks (e.g., procedures). We suspected that the action means-ends/plan-goal relationships captured by PGG would facilitate integration with the structural means-ends work domain relationships captured by the ADS.

Figures 1 and 2 provide the flavor of our analysis. The PGG formalism demands an iterative interleaving of goals and plans or tasks (the terms are used synonymously here) at subsequent levels. Goals define intended states of the world; tasks represent action sequences that are expected to accomplish their parent goals, either individually, collectively or in combinations. PGG endeavors to represent the set of available task methods for accomplishing a goal and has been generally used (Geddes, 1994) to facilitate tracking a user's choice of methods whenever alternatives exist.

We found that the structural means-ends relationships captured in the ADS provided an excellent source of

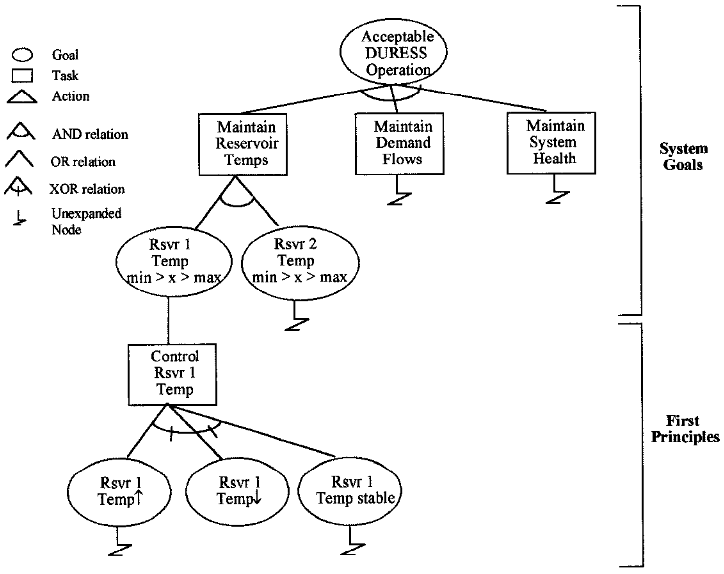

Figure 1. Partial PGG for DURESS II—Top level.

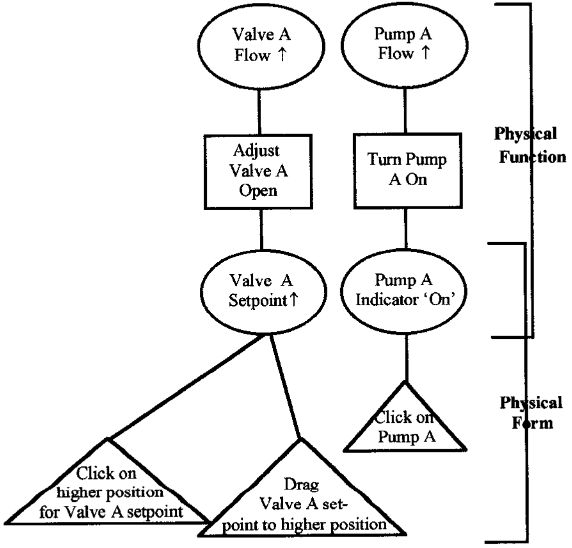

Figure 2. Partial PGG for DURESS II—Bottom/action level.

information about action possibilities around which tasks could be composed or arranged. Each goal in the PGG corresponds to a work domain state represented in the abstraction hierarchy of the ADS, and the action means-ends decomposition of actions in the PGG parallels and is informed by the structural means-ends decomposition of system function represented in the ADS. This is why the progressive layers of the PGG could be mapped loosely into the different layers of the ADS means-ends decomposition, as shown along the right-hand side of each figure. This also corresponds to the intuition that specific work domain states and structural elements are more pertinent to some tasks than to others, and offers a partial explanation for why useful context-sensitive interfaces can be constructed using either a task-based or a system state-based orientation for determining context. Further, through these goal links, the work domain constraint knowledge contained in ADS could be readily used to identify poor or impossible task choices in the PGG. Given governing state equations in ADS, these constraints could even be determined dynamically as world state changes.

Other significant findings from this analysis effort include:

1. The PGG proved surprisingly easy to construct—undoubtedly because the plan-goal relationships that are explicitly modeled in PGG are closely related to the structural means-ends relations that are modeled in ADS. This suggests that a promising approach for linking task knowledge with work domain knowledge is through the integration of their means-ends relationships.
2. The ADS model provided a more rigorous, system-valid basis for building a PGG than might have been available through relying on users' mental models. Years of study with DURESS II have shown that different users develop different strategies for operating the system, and different levels of understanding of its underlying mechanisms (Vicente, 1996). These different 'mental models' of the system provide different levels of support in different usage contexts. If we had been limited to constructing a PGG on the basis of the understanding provided by a small set of these users, the results might have been very different.
3. As a specific example of #2 above, attempting to closely follow the ADS structural means-ends decomposition led to incorporating information into the PGG that probably would have been omitted otherwise. For example, we were tempted to jump immediately from "Control Rsvr1 Temp" to methods of controlling flows and heaters rather than progressing through the Abstract Function layer of the ADS (representing mass and energy balances). That is because generalized mass and energy balance representations are less 'natural' or user-centered, especially for those users of DURESS II who do not have an extensive engineering background. If mass and energy balances are the 'right' or most appropriate way to think about the domain, (as engineering practice and empirical research with DURESS II have shown—Vicente, 1996) then by building PGG tasks around them, we arguably come up with a better abstraction of tasks and goals.
4. On the other hand, the operator task-focus imposed by the PGG demanded some considerations outside the scope of the ADS. The part-whole dimension of the ADS analysis decomposes the overall feedwater system for DURESS II into two subsystems based on physical proximity. While this is a valid representation of the work domain, since each of these subsystems can provide flow into either of two reservoirs, the PGG analysis suggested that a dynamic grouping in terms of those components *currently* contributing to each reservoir's input might be more useful for tasks concerned with individual reservoir flows.
5. Our PGG analysis currently includes no monitoring or decision making tasks. This is likely an artifact of using the ADS as a starting point and, perhaps, of the level of detail to which the analysis was carried. Since this information is ultimately constrained by system performance, it represents an avenue for closer ties between the two techniques.
6. We made no attempt in this analysis to incorporate known procedures or task sequencing knowledge. This emphasized the strengths of the PGG representation, but it undercut one of the primary, complimentary advantages of a task representation vis a vis a work domain representation. Future work will explore extending our PGG representation to include this information.

OUTPUTS FOR INTERFACE DESIGN

Although still very much in the development stages, we can make some claims about the goals of an integrated TA and WDA interface design technique. Such a technique should provide and integrate the information currently available from *both* TA and WDA techniques. Initial results suggest that WDAs tend to identify the information that operators need to monitor and control a system, and thus, the information that must be included in an interface. TAs, on the other hand, tend to identify and prioritize that information, providing guidance to an interface designer in filtering, prioritizing and clustering information.

There is an inherent tradeoff here, of course. As discussed above, one of the strengths of interfaces based on WDAs is their claim to information completeness—that is, if a complete WDA has been conducted, and all relevant work domain constraints and relationships have been identified, then an interface based on this analysis *will* have the information needed to manage the system in any set of circumstances—even those unanticipated by the system designers. On the other hand, there is no guarantee that the operator will be able to process, perceive or understand all of that information simultaneously. The outputs of a TA will provide guidance in how to sort and prioritize that information for presentation, and thus may provide support in minimizing information overload—but only at the cost of failing to present all needed information simultaneously.

In future work, we will be seeking methods by which an integrated task and work domain analysis technique can provide guidance in making that tradeoff. We believe that knowledge about the work domain constraints and capabilities should serve to help identify task possibilities, and that knowledge of an operator's goals or intentions should help identify those specific work domain elements and relationships which are pertinent at any point in time. In any case, as with most current analysis techniques, the results of an integrated TA and WDA analysis will only identify display *requirements*. The creative step of designing graphical elements (or elements in other media) to convey information to and from the operator remains in the hands of an interface designer—albeit a designer now equipped with a more complete understanding of the display requirements driven by both the work domain and by what the operator wants to accomplish in or on it.

FUTURE WORK

In future work, we intend to extend the TA of DURESS II to incorporate sequential task knowledge, begin exploration of the interface design implication of these integrated TA and WDA representations and, as our approach evolves, apply it to a specific work domain suggested by NOVA Chemicals: the management of an acetylene hydrogenation reactor in an ethylene refinery.

ACKNOWLEDGEMENTS

This work has been partially funded by a research grant from NOVA Chemicals Ltd., through the support of the Canadian Natural Sciences and Engineering Research Council, and through internal research support of the Honeywell Technology Center. The authors would like to thank Jamie Errington, Dave Weizenbach, Greg Jamieson and Peter Bullemer for their assistance in developing, steering and conducting this project.

REFERENCES

Bisantz, A. and Vicente, K. (1994). Making the abstraction hierarchy concrete. International Journal of Human-Computer Studies, 40. 83-117.

Booth, P. (1989). An Introduction to Human-Computer Interaction. Lawrence Erlbaum; Hillsdale, NJ.

Diaper, D. (1989). Task analysis for human-computer interaction. Ellis Horwood; Chichester, UK.

Funk, H. & Miller, C. (1997). Task sensitive versus 'context sensitive': What should an adaptive interface adapt to? In Proceedings of the International and Interdisciplinary Conference on Modeling and Using Context. Rio de Janeiro; February 4-6.

Geddes, N. (1994). A model for intent interpretation for multiple agents. In Proceedings of the IEEE conference on Systems, Man and Cybernetics. October; Houston, TX.

Gibson, J. J., & Crooks, L. E. (1938). A theoretical field-analysis of automobile-driving. American Journal of Psychology, 51, 453-471.

Kirwan, B. and Ainsworth, L. A Guide to Task Analysis. Taylor and Francis; Bristol, PA.

Miller, C., Funk, H., and Hannen, M. (1997). "Task-Based Interface Management; A Rotorcraft Pilot's Associate Example." In Proceedings of the American Helicopter Society's Crew Systems Technical Specialists Meeting. Philadelphia, PA; September 23-25.

Norman, D. (1989). The Design of Everyday Things. Doubleday; New York.

Rasmussen, J. (1985). The role of hierarchical knowledge representation in decision making and system management. IEEE Transactions on Systems, Man and Cybernetics, 15, pp. 234-243.

Rasmussen, J., Pejtersen, A., and Goodstein, L. (1994). Cognitive Systems Engineering. John Wiley & Sons; New York.

Rouse, W., Geddes, N. and Curry, R. (1988). An Architecture for intelligent interfaces: Outline of an approach to supporting operators of complex systems. Human-Computer Interaction, 3. pp. 87-122.

Sewell, D. and Geddes, N. (1990). A plan and goal based method for computer-human system design. In D. Diaper (Ed.) Human-Computer Interaction—INTERACT '90. Elsevier Science; North-Holland. pp. 283-288.

Vicente, K. J. (1996). Improving dynamic decision making in complex systems through ecological interface design: A research overview. System Dynamics Review, 12, 251-279.

Vicente, K. J. (in press). Cognitive work analysis: Towards safe, productive, and healthy computer-based work. Erlbaum: Mahwah, NJ.

Vicente, K. J., & Rasmussen, J. (1992). Ecological interface design: Theoretical foundations. IEEE Transactions on Systems, Man, and Cybernetics, SMC-22, 589-606.

Wickens, C. (1992). Engineering Psychology and Human Performance (2nd ed.). Harper-Collins; New York.

Woods, D. (1986). Paradigms for intelligent decision support. In E. Hollnagel, G. Mancini, and D. Woods (Eds.), Intelligent Decision Support in Process Environments. Springer-Verlag; New York.

Woods, D. (1991). The cognitive engineering of problem representations. In G. Weir and J. Alty (Eds.) Human-Computer Interaction and Complex Systems. Academic Press: London.