Ecological Interface Design
Theoretical Foundations

Vicente, Kim J.; Rasmussen, Jens

Published in:
IEEE Transactions on Systems, Man and Cybernetics

Link to article, DOI:
10.1109/21.156574

Publication date:
1992

Document Version
Peer reviewed version

Citation (APA):
Vicente, K. J., & Rasmussen, J. (1992). Ecological Interface Design: Theoretical Foundations . IEEE Transactions on Systems, Man and Cybernetics, 22(4), 589-606. DOI: 10.1109/21.156574

ECOLOGICAL INTERFACE DESIGN: THEORETICAL FOUNDATIONS

Kim J. Vicente and Jens Rasmussen[1]

ABSTRACT

A theoretical framework for designing interfaces for complex human-machine systems is proposed. The framework, called ecological interface design (EID), is based on the skills, rules, knowledge taxonomy of cognitive control. The basic goal of EID is twofold: first, not to force processing to a higher level than the demands of the task require, and second, to support each of the three levels of cognitive control. Thus, an EID interface should not contribute to the difficulty of the task, and at the same time, it should support the entire range of activities that operators will be faced with. Three prescriptive design principles are suggested to achieve this objective, each directed at supporting a particular level of cognitive control. In this paper, the theoretical foundations of the framework are laid out. Particular attention is paid to presenting a coherent deductive argument justifying the principles of EID. In addition, three sources of converging support for the framework are presented. First, a review of the relevant psychological and cognitive engineering literature reveals that there is a considerable amount of research that is consistent with the principles of EID. Second, an examination of other approaches to interface design indicates that EID has a unique and significant contribution to make. Third, the results of an initial empirical evaluation also provide some preliminary support for the EID framework. The paper ends by outlining some issues for future research.

1. INTRODUCTION

This paper describes a novel theoretical framework for interface design for complex human-machine systems. The framework, called ecological inter-

---

[1] Kim J. Vicente is with the Center for Human-Machine Systems Research, School of industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, GA 30332-0205.
Jens Rasmussen is with the Technical University of Denmark and the Cognitive Systems Group, Riso National Laboratory, DK-4000 Roskilde, Denmark.

face design (EID), is an attempt to extend the benefits of direct manipulation interfaces (DMI) to complex work domains. Unfortunately, existing theories of DMI [41, 97] were not specifically developed with complex human-machine systems in mind. As a result, it should not be surprising to find that these theories do not effectively address the unique challenges posed by complex work domains [88]. The first step towards developing a more appropriate design framework, then, is to identify the important challenges associated with the design of complex human-machine systems. The discussion will take place within the specific context of process control, although the issue of generalizability to other work domains will be addressed later in the paper.

A. Unanticipated Events

One way to classify events in complex human-machine systems is according to their degree of novelty from the perspective of first operators and then designers. Three broad areas along a continuum can be identified:

1. Familiar events are routine in that operators experience them frequently. As a result of a considerable amount of experience and training, operators have acquired the skills required to deal with these events.
2. Unfamiliar, but anticipated events occur infrequently and thus operators will not have a great deal of experience to rely on. However, the events have been anticipated by plant designers, who have built in means to deal with them (e.g., procedures, decision support systems, automatic controllers, etc.). These anticipated solutions provide operators with the help they need to cope with this class of events.
3. Unfamiliar and unanticipated are also unfamiliar to operators because they rarely occur. Unlike the previous category, however, the event has not been anticipated by designers. Thus, operators cannot rely on a built in solution but must improvise one themselves.

Ideally, one would like to alleviate the burden on operators by designing plants in such a way that unanticipated events will not occur. However, there is a consensus that this is not a technically feasible option [54,55,104]. As the Three Mile Island (TMI) Lessons Learned Task Force pointed out [104], the set of events that is used as a basis for design does not constitute an exhaustive list of events that can occur. Rather, it consists of a number of representative classes of scenarios that are judged by designers to be of sufficient likelihood and severity that they are worthwhile considering (e.g., [42]). Consequently, there will always be a chance that significant, safety threatening events have been overlooked and therefore not included in the set of design basis events. The inescapable conclusion is that unanticipated events can and do happen in large-scale industrial systems. In fact, they are the major cause of life-threatening accidents [73,89].

What types of human factors problems are posed by unanticipated events? While human performance in routine events is primarily susceptible to alil?i (i.e., errors of execution) (cf. [67]), performance in unanticipated events is limited more by mistakes (i.e., errors of intention). The frequency of slips can be reduced through the application of traditional ergonomic guidelines; mistakes, on the other hand, can only be prevented by considering the cognitive factors influencing operator behavior [1 16]. It follows, therefore, that the demands posed by unanticipated events cannot be overcome simply by designing an interface with well laid out and clearly labelled controls and displays. Yet much of the work that has been done on interface design for complex systems has focussed on these types of ergonomic issues rather than on semantic issues (e.g., [47, 1021).

As a result, it should not be surprising to find that traditional interface design practices do not provide operators with the support they need for coping with this class of events. The TMI accident, for instance, made it patently clear that traditional control rooms provide operators with inadequate information to reflect plant status under unanticipated event sequences [103]. Subsequent efforts to identify the information set needed by operators to deal with off-normal events have tended to adopt a common approach. A set of events is selected, and then the information needed to diagnose each event sequence is determined (e.g., [47, 102]). But this type of procedure for determining what information should be included in an interface cannot, by definition, cope with unanticipated events. Clearly, an alternative approach is required.

B. Outline

This paper will develop a framework for interface design that attempts to support operators during familiar, unfamiliar, and in particular, unanticipated events. The structure of the paper is as follows. First, a generic structure describing the interface design problem is defined. To anticipate, this problem formulation leads to two questions that must be addressed by a design framework, specification of information content and design of visual form. In the next section, the abstraction hierarchy is proposed as a psychologically relevant framework that allows one to specify the information content of an interface in a way that provides operators with a basis for coping with unanticipated events. Then, the skills, rules, knowledge taxonomy is used to integrate_ a variety of findings from the literature so as to derive inferences for how information should be presented in an interface. These theoretical developments provide the justification and context for the description of the three principles composing the EID framework. A review of other approaches to interface design is then undertaken to determine the contribution of the EID framework. Finally, the results of an initial empirical evaluation of one of the principles of the EID framework is also briefly presented.

C. Relation to Previous Work

The EID framework was first proposed in [88]. The goal of that paper was to describe the relationship between different classes of errors and the implications those errors had for interface design. In a subsequent paper f 107], a detailed design example for a therrnal-hydraulic process simulation was presented, showing how the principles of EID can be applied in a prescriptive manner to develop a concrete design product. In addition, the conceptual ties between E]ID and ecological psychology were also pointed out at length. The emphasis of the present paper is on making explicit, in detail, the rationale behind the principles of EID by reviewing the theoretical and empirical evidence pertaining to the problems addressed by the framework. Therefore, in addition to presenting a coherent deductive argument justifying the EID principles, the present paper also puts the EID framework within a more global context defined by psychological and cognitive engineering literature. Those interested in a detailed example of how the ideas presented here can be applied are referred to [107].

II. PROBLEM FORMULATION

This section is directed at formulating the problem of interface design from a research perspective. The goal here is not to develop design principles (this will be addressed later), but to first structure the problem in a meaningful way. The resulting structure will define the set of questions that must be addressed by the framework to be developed in the remainder of the paper.

A. Fundamentals

An interface is part of a control system involving human and machine components. The operation of any control system is bound by the laws of control theory, which provides a generic language for describing such systems. Accordingly, it is important to consider the various fundamental constraints that the discipline of control theory places on systems design. First, the Law of Requisite Variety [2] states that complex systems require complex controllers. It follows, then, that in designing a human-machine system, the complexity inherent in the process cannot be displaced if effective control is to be achieved. That complexity must be dealt with in one way or another, whether it be by the designer, the machine, or the operator. Second, it is important to realize that physical systems can be described by a set of constraints. Examples of possible sources of constraint include the purposes for which the system was designed, natural laws governing system behavior, organizational or legal policies, the nature of the functions that have been built into the system, the characteristics of the equipment available to carry out those functions, and the spatial topology and appearance of the system components. Third, it is well known in linear systems theory that, implicitly or explicitly, every good controller must be, or possess, a model of the system it is controlling [14, 24]. These last two points imply that, in order to properly control the process, the human-machine system must take into account, or embody, the constraints inherent in the work domain [101]. In other words, optimal control requires a consideration of veridical system functioning.

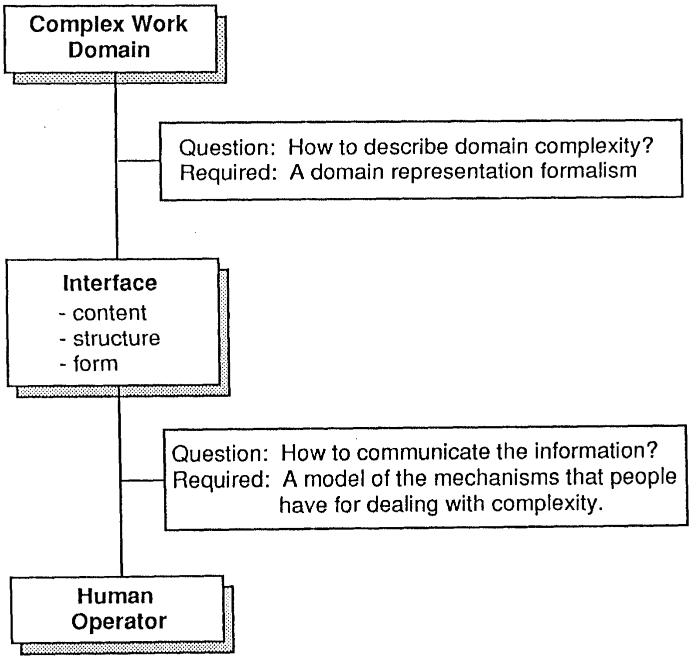

Figure 1. The structure of the interface design problem.

B. The Structure of the Design Problem

As shown in Figure 1, two questions pertinent to interface design arise from these fundamental considerations. First' what is a psychologically relevant way of describing the complexity (i.e., variety [2]) of the work domain? This requires a representation formalism for describing the work domain's constraints. Such a representation defines the informational content and structure of the interface. Second, what is an effective way of communicating this information to the operator? Here, an understanding of the mechanisms that people have available for processing information is required. This would provide a basis for determining the form that the information should take, the idea being that information should be presented in a form that is compatible with human cognitive and perceptual properties. These two questions define the core of the interface design problem.

It is interesting to note that one of these questions is primarily related to the characteristics of the domain, whereas the other is related to those of the operator. This structure is very similar to the organism-environment reciprocity that is central to ecological psychology [8, 9, 27, 28]. In fact, this duality between operator and work environment was one of the reasons for calling this theoretical framework "ecological" (see [88] and especially [107] for a much more comprehensive discussion of the numerous parallels between ecological psychology and E]ID).

Of course, there are other questions pertinent to interface design, such as issues of context sensitivity [63, 113], visual momentum [1 12-1, and dialogue [109]. These are all important facets of interface design that are not addressed by EID. Thus, in its present form, the scope of EDD is limited to the most basic functions of an interface, as laid out in Figure 1. These are the core problems that must be addressed. The other issues just described address the additional roles that an interface must play in complex human-machine systems. Ideally then, work on these other problems should be integrated with the EID approach to provide a comprehensive approach to interface design.

In the following sections, two theoretical concepts are adopted to answer the two questions just posed. First, the abstraction hierarchy [77, 78, 80] is proposed as a psychologically relevant form of representing the constraints in a work domain in a way that allows operators to cope with unanticipated events. Second, the skills, rules, knowledge (SRK) taxonomy [75, 77, 79] is proposed as a useful framework for describing the various mechanisms that people have for processing information. The goal is to adopt these two theoretical constructs as axioms and then to use them to derive a number of implications for interface design.

III. THE ABSTRACTION HIERARCHY

The abstraction hierarchy is a useful framework for representing a work domain in a way that is relevant to interface design. The goal of this section is to review the evidence supporting this claim. First, the structure of the abstraction hierarchy will be described. Second, an argument showing how an interface based on an abstraction hierarchy representation provides operators with a basis for coping with unanticipated events will be put forth.

Third, other research that relies on a set of insights similar to those on which the abstraction hierarchy is based will be briefly reviewed. Fourth, evidence showing that the abstraction hierarchy provides a psychologically relevant representation for problem solving will also be reviewed. In the ensuing discussion, it is assumed that the reader has at least some familiarity with the arguments presented in [78].

A. What Kind of Hierarchy?

Different types of hierarchical structures have frequently been used to represent complex systems in a variety of disciplines (e.g., [1, 50, 58, 69, 98]). However, there are certain properties which distinguish the abstraction hierarchy from other types of hierarchies. To better understand the distinctive properties of Rasmussen's abstraction hierarchy (and its corresponding benefits), the general class of hierarchies of which the abstraction hierarchy is a subclass must first be defined.

The abstraction hierarchy belongs to the class of stratified hierarchies described by Mesarovic, Macko, and Takahara [58], the properties of which are listed below.

1. Each stratum, or level, of the hierarchy deals with the very same system, the only difference being that different strata provide different descriptions, or different models for observing the system.
2. Each stratum has its own unique set of terms, concepts, and principles.
3. The selection of strata for describing a particular system depends on the observer, and his knowledge and interest in the control of the system. For many systems, however, there may be some strata which appear to be natural or inherent.
4. The requirements for proper system functioning at any level appear as constraints on the meaningful operation of lower levels, while the evolution of the state of the system is specified by the effect of the lower levels on the higher levels.
5. Understanding of the system increases by crossing levels: by moving up the hierarchy, one obtains a deeper understanding of system significance with regard to the goals that are to be achieved, while in moving down the hierarchy, one obtains a more detailed explanation of the system's functioning in terms of how those goals can be carried out.

In addition to these characteristics, the structure of the abstraction hierarchy is further specified by a means-end relationship between levels [78]. This is in contrast to other types of hierarchies which are often defined by attributes which are not explicitly related to goals (e.g., spatial scale, temporal scale, authority, flow of information). This explicitly goal-oriented nature has important psychological implications (see below).

Note that the properties just described define a family of representations. In other words, the abstraction hierarchy is not a specific representation but rather a framework for developing representations for various work domains. The exact number of levels and their content will vary from domain to domain as a function of the different types of constraints inherent in each work domain. For example, five levels of constraint have been found to be useful for describing process control systems [781: the purposes for which the system was designed (Functional Purpose); the intended causal structure of the process in terms of mass, energy, information, or value flows (Abstract Function); the basic functions that the plant is designed to achieve (Generalized Function); the characteristics of the components and the connections between them (Physical Function); and finally, the appearance and spatial location of those components (Physical Form). Regardless of the domain, however, the resulting representation will have the properties just described (see also [78]).

What unique advantages are offered by a representation with such characteristics? The remainder of this section will address this question. To anticipate, an abstraction hierarchy representation has two important benefits: it provides operators with an informational basis for coping with unanticipated events, and it provides a psychologically valid representation for problem solving. Each of these topics will now be addressed in turn.

B. Coping with Unanticipated Events: A Historical Overview

An understanding of how the abstraction hierarchy provides a basis for coping with unanticipated events is best acquired by examining its historical origins. Because this information can only be found in technical reports which are not widely known or available, a detailed discussion is warranted. Although the abstraction hierarchy first appeared in a technical report published in 1979 [80], it has its origins in a research program on complex human-machine systems that began in the 1960's at Riso National Laboratory in Roskilde, Denmark. This research originated with the concern of analyzing and improving the reliability and safety of complex, industrial systems, particularly nuclear power plants (NPP's).

Early efforts were directed at analyzing the reliability of plant equipment and instrumentation [44, 87]. Further investigation, however, lead to the realization that the reliability of such systems cannot be viewed strictly from a technical viewpoint without considering the human element in the system [84, 85]. It became apparent that the human operator plays a key role in overall system reliability and safety. This observation was supported by reviews of 29 cases with major consequences to either plant or personnel in the nuclear domain and of 100 accidents in air transportation [82]. The results of this analysis indicated that accident-causing errors arose because human operators were confronted with unfamiliar situations which had not been, or could not have been, anticipated by designers. In contrast, under normal circumstances, a trained and experienced operator will often be able to compensate for deficiencies in the interface [82, 83]. Consequently, the single most important concern in improving system safety is to provide operators with the support required to deal with unfamiliar and unanticipated abnormal situations. The subsequent research program that was undertaken at Riso was primarily directed at developing a design framework to deal with this challenging problem.

The first step taken in this direction was an engineering analysis of the control requirements posed by unanticipated events [8 1]. Looking at the problem at a fundamental level revealed several important insights. First, when the system is functioning correctly, it can be described by a set of constraints that are imposed on the observed data set by the functioning and anatomy of the system being controlled. Because the system was built in a certain way, for a certain purpose, there will be certain relationships between variables. These relationships can be described as constraints. These constraints represent "rules of rightness" [74], or in the language of Mesarovic et al. [58], it goals to be achieved"; to say that the system is operating normally is equivalent to saying that the constraints in question hold. When a fault occurs, however, system structure and functioning will change. This means that the system is no longer governed by the same constraints. Consequently, an abnormality results in the breaking of one or more constraints that govern the system under normal circumstances. From this perspective, the task of fault detection is equivalent to detecting the breaking of constraints. However, to be able to detect such a change, the states of all of the variables entering into the violated constraint must be represented, otherwise it will not be possible for the operator to uniquely determine if a constraint has indeed been broken. The problem is compounded by the fact that it is not possible to know beforehand which constraint will be violated. The implication for interface design is that the complete set of goal-relevant constraints governing the system must be represented to permit operators to determine when a constraint has been broken, and thereby allow them to directly diagnose the abnormality. (For a formal instantiation of this argument within the context of a specific process control system, see [106]).

These insights eventually lead to the abstraction hierarchy, which provides a framework for identifying and integrating the set of goal relevant constraints that are operating in a given work domain. Each level in the hierarchy represents a different class of constraint, or in the terms of Mesarovic et al. [581, a different stratum (see the properties of stratified hierarchies listed in the previous subsection). One way to think of the abstraction hierarchy, then, is as a set of models of the system, each defining a level of the hierarchy [80]. Higher levels represent relational information about system purpose, whereas the lower levels represent more elemental data about physical implementation.

With respect to interface design, the important implication is that because higher order, functional relations are explicitly represented, it should be possible for operators to determine when process constraints are broken. (The qualifier is required because making the necessary information available does not guarantee that it will be attended to or interpreted correctly). Consequently, developing an abstraction hierarchy for a work domain allows designers to identify the information that operators need to cope with the entire range of operating demands, including unanticipated events.

C. Relation to Other Work

It is worthwhile pointing out that the insights behind the abstraction hierarchy have been independently recognized and adopted by other researchers. For example, Davis [15] discusses the need for multiple representations defining different types of constraints in the context of troubleshooting. He describes an artificial intelligence program for troubleshooting digital electronic circuits that reasons from first principles. Rather than relying on a list of previously enumerated faults, Davis' program detects a mismatch between values expected based on knowledge of proper system functioning and those actually obtained. This allows the system to diagnose events which have not been explicitly built into the system's knowledge-base and which have not been encountered previously.

The general idea of exploiting redundant functional relations for diagnosis has also been recognized by control theorists (see [25] for a review). The goal is to use an analytical model of the system as a referent for proper system functioning, comparing this referent to the current state of the system, and then analyzing the residual for an unexpected deviation signifying a fault. The difference of course is that the approach suggested here merely attempts to provide the operator with the informational basis for performing the diagnosis. Responsibility for detection, diagnosis, and compensation is left in the hands of the operator. In contrast, control theorists have attempted to design automated systems to perform all of these activities.

D. Psychological Relevance

From an engineering perspective, an interface based on an abstraction hierarchy representation has the benefit of providing an informational basis for coping with unanticipated events. However, there are also psychological justifications for adopting the abstraction hierarchy as a basis for interface design.

One important property of an abstraction hierarchy representation is that higher levels are less detailed than lower levels. This fact has important psychological implications. Shifting one's representation from a low (i.e., very detailed) level to a higher level of abstraction with less resolution makes complex systems look simpler. In effect, this provides a mechanism for coping with complexity [86]. Metaphorically, moving up one or more levels allows one to "see the forest through the trees". Thus, part of the psychological relevance of the abstraction hierarchy lies in the fact that it allows resource-bounded agents, as people are, to deal with systems that would be unmanageable if they had to observe the whole system in full detail all at once.

This advantage is not unique to the abstraction hierarchy, however (cf. [98]). Most, if not all, hierarchies allow one to observe systems at a less detailed level. From a psychological point of view, the unique and important characteristic of the abstraction hierarchy is that it is explicitly goal-oriented. The various levels in the hierarchy are linked by a means-end relation.[2] This relationship provides a very important source of constraint that can be exploited in problem solving. Thus, search can be constrained by initiating the problem solving process at a high level of abstraction, deciding which part of the system is relevant to current goals, and then concentrate on the sub-tree of the hierarchy that is connected to the subsystem of interest. To take a concrete example from electronic troubleshooting, one could start off by examining a television at a relatively coarse level of description, such as generic functions. One could then identify the faulty function (e.g., the power supply) and then drop down to the next level to look at only those components that are functionally connected to the power supply. This is an efficient form of search since all components not pertinent to the power supply can be ignored (see Korf [50] for a formal treatment of the computational efficiency of this type of constrained search). In summary, an abstraction hierarchy representation allows one to engage in goal-directed problem solving in a computationally economic manner.

Note that this advantage is not enjoyed by other types of hierarchical representations (e.g., partwhole hierarchies, or classification hierarchies). With

---

[2] Many studies in the cognitive science literature, conducted in quite disparate domains, have observed that the knowledge representation of experts is organized in a functional hierarchy [12, 30]. Since the abstraction hierarchy is in fact a "functional hierarchy", one could conceivably reinterpret this generalization as indicating that experts' knowledge structures in various domains are organized according to an abstraction hierarchy. However, this would be going well beyond the data because, unfortunately, the precise structure of the "hierarchical" organization observed in these studies has typically not been well defined, neither within nor across domains (e.g., [I 3]). The abstraction hierarchy, defined as a stratified hierarchy with a means-end relations between levels, can be viewed as one attempt at making more explicit exactly what a "functional hierarchy" might look like. Future research should determine whether this more precise definition of "functional hierarchy" can indeed capture the structure of experts' knowledge representations in various domains.

these other representation fornnats, the links between levels are not necessarily related to goals. While it is still possible to examine the system at a high level of the hierarchy to choose a subsystem of interest, the critical point is that the sub-tree of the hierarchy that is connected to that subsystem may not necessarily contain system components that are relevant to the goals that the selected subsystem is designed to achieve. In other words, other hierarchies constrain search but not in a way that is explicitly related to the purposes for which the system is designed. It is this latter type of constraint that is needed for facilitating goal-directed behavior.

If people do indeed reason within an abstraction hierarchy representation, then this should reveal itself in several empirically observable ways. First, it should be possible to meaningfully map problem solving protocols onto an abstraction hierarchy representation of the domain. Several studies show that this is possible. In fact, there is a body of empirical literature leading from Selz's [96] seminal work (see [26] for an English account), which illustrates the psychological validity of the abstraction hierarchy as a problem space representation (see [77] for a detailed review). For instance, Duncker [20] found that verbal protocols of subjects solving practical problems could be mapped onto an abstraction hierarchy representation. The same observation was made by de Groot [17], who collected verbal protocols of the problem solving activities of world-class chess players. More recently, the problem solving behavior of expert computer programmers has also been found to be consistent with search through an abstraction hierarchy space [1051. Also, Rasmussen discovered that the problem solving activities of professional troubleshooters performing real diagnostic tasks could also be mapped onto an abstraction hierarchy of the equipment being repaired (see [77, p. 1191 for a specific example). Finally, Itoh, Yoshimura, Ohtsuka, and Masuda [43] found that the problem solving behaviors of NPP operators could be mapped onto an abstraction hierarchy representation of the plant (see [43, p.101] for a specific example).

Second, one would also expect that subjects' problem solving trajectories would begin at a high level of abstraction and gradually focus in on lower levels, thereby exploiting the goal-relevant constraint provided by the hierarchy. This type of "zooming in" behavior has indeed been observed in virtually each of the studies just cited. These observations are also consistent with recent research on problem solving expertise which has consistently shown that experts spend a great deal of their time analyzing the functional structure of a problem at a high level of abstraction before narrowing in on more concrete details [30].

In summary, an argument was put forth to show that, in contrast to hierarchies which are not defined by a means-end relation, the abstraction hierarchy allows one to constrain goal-directed search in a computationally economic manner. Furthermore, the empirical evidence indicates that the abstraction hierarchy can meaningfully capture the richness of problem solving protocols from complex and practically meaningful activities in a variety of domains. Thus, there is evidence to indicate that the abstraction hierarchy is a psychologically plausible problem solving representation.

E. Conclusion

To conclude, there is converging evidence that the abstraction hierarchy is a useful way to represent a work domain. Not only is it a psychologically relevant problem representation but it also provides operators with an informational basis for coping with unanticipated events which, as pointed out in the introduction, are a major threat to safety in complex human-machine systems. This problem representation can provide the foundation for interface design by specifying the information content and structure of the interface (see [107] for an example).

The next question is: What mechanisms do people have to cope with the complexity inherent in the domain? Answering this question will allow one to detem-iine effective ways of communicating the information in the representation to the operator (the second design problem shown in Figure 1). This will be the topic of section IV.

To anticipate, the strategy adopted here is to take advantage of the most powerful resources that people have for dealing with complexity. As Newman [66] has noted, "People don't mind dealing with complexity if they have some way of controlling or handling it .... if a person is allowed to structure a complex situation according to his perceptual and conceptual needs, sheer complexity is no bar to effective performance" (p. 9). In order to adopt this approach, one needs to know something about the different mechanisms that people have for processing information, how these can be induced, and what their relative efficacy is. This set of issues will be addressed next with the help of the SRK taxonomy.

IV. MULTIPLE LEVELS OF COGNITIVE CONTROL

In the systems reliability and cognitive engineering communities, the SRK taxonomy [79] has become a widely accepted framework for describing the various mechanisms that people have for processing information [90, 94]. The basic tenets of the taxonomy are that information can be interpreted in three mutually exclusive ways -- as signals, signs, or symbols -- and that the way in 2 which information is interpreted determines which of the three levels of cognitive control[3] is activated -- skill-based behavior (SBB), rule-

---

[3] Rasmussen's taxonomy of cognitive control can be viewed as a hierarchical control system consisting of three levels. The knowledge-based level is at the top, the rule-based level is in the middle, and skill-based behavior is the bottom level. Those not familiar with the taxonomy are refer-red to [79] for a more detailed description. Note that cognitive control,

based behavior (RBB), and knowledge-based behavior (KBB), respectively. Thus, cognitive control may depend on a repertoire of automated behavioral patterns (SBB), a set of cue-action mappings (RBB), or problem solving operations on a symbolic representation (KBB).

In this section, the SRK taxonomy will be adopted as an "umbrella" for integrating a variety of research results under a common language. The theoretical constructs of SRK will then be used to make certain deductions from these findings, which in turn, will lead to a specific set of recommendations for interface design.

A. The Power of Perception

The three levels of cognitive control can be grouped together into two general categories (cf [89]). KBB is concerned with analytical problem solving based on a symbolic representation, whereas RBB and SBB are concerned with perception and action. The distinction between these two modes of processing is common to most, if not all, human performance frameworks [90]. In general, perceptual processing is fast, effortless, and proceeds in parallel, whereas analytical problem solving is slow, laborious, and proceeds in a serial fashion. Furthermore, because of working memory limitations, analytical problem solving also tends to be more error-prone than perceptual processing [89]. It is important to note, however, that the lower levels can only be activated in familiar situations because they require that the operator be attuned to the perceptual features of the environment. KBB, on the other hand, allows operators to cope with novelty. Thus, this dual cognitive architecture allows people to trade off processing efficiency for the ability to deal with unfamiliar events. No level is globally superior to any other.

There are two characteristics of complex work domains which make it possible to apply this knowledge to design. First, operators of such systems are highly skilled and have extensive experience in controlling the system. Second, interface design for complex systems consists of specifying an interface for a single, specific application; generality is not important. Thus, issues associated with transfer between various applications do not play a significant role because operators will almost always be dealing with the same interface for the same process.

These two factors make perceptual processing (i.e., SBB and RBB) an attractive possibility. Since operators will have extensive experience with the system, they will have the opportunity to attune themselves to the perceptual properties of the control room interface. Also, the fact that operators will be dealing with the same process means that, if a way can be found to

---

as used here, is entirely unrelated to the definition of cognitive control provided by Hammond and Summers [361.

comprehensively describe the work domain, then the need for dealing with novel situations will be minimized. This, in turn, implies that the reliance on KBB should also be reduced. Together, these two characteristics of complex systems suggest that interfaces should be aimed at taking advantage of the processing efficiency of lower levels of cognitive control.

Is there any empirical evidence to support this reconunendation? While many researchers have argued for the immense power of people's perceptual abilities (e.g., [9, 16, 18, 27, 28, 48, 49, 77, 89, 91]), only a few studies have ever directly compared perception and analytical reasoning. Brunswik [9, pp. 89-93] seems to have been the first to empirically address the relativeefficacy of what he referred to as perception (SBB and RBB) and thinking (KBB). He presented subjects with a size constancy task in two different forms, one requiring perception and the other requiring arithmetic reasoning. The results indicated that in the perceptual version of the task, subjects'responses were centered around the correct answer with a relatively small degree of variability. In contrast, with the analytical version of the task, more subjects reported the precise correct answer than in the perceptual version, but the standard deviation in performance was more than ten times that obtained for the perceptual version! Thus, whereas perception was rarely perfect but always close, thinking could be perfect but sometimes led to extreme errors. Given that perception is faster than thinking, Brunswik [9] concluded that: "die balance sheet of perception versus thinking may thus seem seriously upset against thinking, unquestioned favorite of a culture of rational enlightenment as the latter has been" (p. 93).

A second direct comparison of perception and analytical reasoning was conducted by Hammond, Hamm, Grassia, and Pearson [37], who refer to the two modes as intuition and analysis, respectively. They conducted an experiment with expert highway engineers on three different judgements tasks presented in various forms so as to differentially induce perception and analysis. Interestingly, Hammond et al.'s [37] results replicated Brunswik's [91 finding that analytical cognition can lead to extreme errors. Moreover, when the effects of nonsystematic errors were removed, the results indicated that perception was frequently superior to analytical thinking in terms of the empirical accuracy of judgements. In a related study investigating the interplay of perception and analysis over time, Hamm [35] also found that perception was more closely related to good performance than analytical cognition.

While these results show that perception can be very effective, as the authors themselves have pointed out, one must be very careful in generalizing the findings. In particular, it is important to realize that relying on lower levels of cognitive control may not always lead to superior performance. The claim being made here is not that perception is always better than analysis but that the conditions characteristic of complex work domains are propitious for perceptual processing and that designers should take this into account in designing interfaces.

B. The Propensity for Perceptual Processing

The discussion so far suggests that it would be highly adaptive to take advantage of the efficiency of perceptual processing. Interestingly enough, people naturally adopt such a strategy. That is, people attempt to simplify complex tasks by taking advantage of their most powerful cognitive resource. In this subsection, a variety of studies supporting this claim will be reviewed (see also [81, 89, 91, 92]).

1) Two examples. The work of Klein [49] provides a good example of people's propensity for perceptual processing. He and his colleagues have conducted a series of naturalistic studies of expert decision making in the domains of fire fighting, military operations, and engineering design. The data were collected by, first, identifying non-routine events requiring skilled decision making, and second, conducting interviews to probe these events in order to examine the nature of the decision making process. Over a hundred cases were analyzed.

Since the incidents examined were non-routine, one would expect that decision making would be, to use Klein's terms, analytical rather than recognitional (i.e., based on KBB rather than RBB). Surprisingly, the results indicated that, even in such critical incidents, experts often relied on the recognitional mode of decision making. Such a strategy is adaptive in several respects. In terms of mental effort, the recognitional mode is less taxing than the analytical mode. In terms of effectiveness, recognitional decision making allows experts to take advantage of their experience. Because of their wealth of experience, experts are able to quickly generate a plausible action alternative, rather than generate the complete set of possible alternatives, as analytical decision making models would suggest. Finally, in terms of appropriateness, recognitional decision making is much quicker than analytical decision making, and therefore allows experts to effectively cope with time stress.

Kirlik [48] has also argued that skilled performance relies heavily on perceptual processing. He had subjects perform a complex, supervisory control task until they became proficient and then tried to model their performance. Given the complexity of the task demands, one would perhaps expect that subjects would have to engage in a great deal of analytical problem solving to perform well. Surprisingly, Kirlik [48] found that it was possible to account for expert behavior by developing a parsimonious model that relied almost exclusively on perception and action. Cognitive processing was only required on the rare occasions when there was not enough information available in the environment to uniquely select an appropriate action. Kirlik's findings lend plausibility to the idea that skilled performers tend to rely heavily on lower levels of cognitive control.

These results, obtained with realistically complex tasks, reinforce the argument outlined in the previous section. If one can design interfaces that allow people to effectively take advantage of perceptual processing, then the benefits can be great. However, there is a major difference between the domains that Klein and Kirlik investigated and those that are of concern here: in complex, high technology systems, the goal-relevant properties of the work domain typically cannot be directly observed by the unaided eye. While the results cited above illustrate the proficient level of performance that can result from exploiting lower levels of cognitive control, they do not provide any indication as to how to derive those benefits through proper interface design, nor do they reveal what can go wrong if the proper support for lower levels is not provided.

2) More examples and what can go wrong. The problem with many existing interfaces is that they penalize, rather than support, operators' preference for lower levels of cognitive control. An experimental study conducted by Hollnagel [39] in the area of process control serves as an excellent example.

Since the process being controlled is not directly observable, there are two phenomenologically different types of control strategies that can be adopted by operators. Following Hollnagel [39], these will be referred to as surface control (corresponding to SBB and RBB) and deep control (corresponding to KBB). Surface control is guided by the perceptual properties of the displays, whereas deep control of the system is guided by the operator's mental model of the underlying process. While this surface/deep control distinction is best thought of as a continuum, studies of process control environments have often indicated that operators have a distinct preference for surface control rather than deep control of the system (e.g., [81, 92]). In other words, process control operators have a preference for lower levels of cognitive control.

Hollnagel's [39] experiment provides a typical example of this pattern of behavior. In his study, subjects tended to disregard the (abstract) functional properties of the process being controlled, and relied on the (concrete) perceptual characteristics of the display instead. In effect, they often treated the process as if it was physically structured as the display indicated. However, as is the case with most existing process control interfaces, the displays were not designed to be complete, veridical representations of the process. In other words, there is no direct, consistent relationship between the perceptual characteristics of the display and the constraints describing process behavior. Thus, it is difficult for operators to consistently control the system by considering the surface features of the display alone. As a result, the tendency toward surface control results in several classes of problems. First, it is easy for operators to forget, and therefore fail to consider, properties of the process which are not shown in the display. Second, because of the inconsistent mapping between the invariant properties of the process and the signs provided by the display, the cues that operators normally rely on to control the system are imperfectly correlated with the state of the system. Thus, in novel situations, surface control will result in under-specification [89] and human error [88] (see section V for an example). Third, if functional relationships between subsystems are not explicitly represented in the display, operators tend to treat the subsystems as being independent of each other. This tendency is enhanced if the two related subsystems are spatially distant from one another (e.g., [93]). These findings show how operators' propensity for lower levels of cognitive control can lead to errors if the interface is not designed with this knowledge in mind.

This preference for lower levels of cognitive control is by no means limited to process control systems. In a study investigating subjects' ability to estimate failure probabilities from a fault tree diagram, Fischoff, Slovic, and Lichtenstein [22] found the same pattern of behavior, which they labelled as "out of sight, out of mind". Subjects who were presented with diagrams that did not contain all possible fault categories tended to ignore the categories not represented. This tendency persisted, although to a lesser degree, when subjects were explicitly told to consider events that were not represented in the fault tree. The results also revealed that events were perceived as more important when they were represented as two branches in the fault tree diagram than when presented as one. One plausible interpretation of these findings is that subjects were basing their judgements on the perceptually salient features of the diagram instead of on their conceptual knowledge of the problem.

The findings of Hollnagel [39] and Fischoff et al. [22] have been replicated by Smith [1001 in the domain of management decision making. His study compared the performance of two groups of subjects, one that was given a deliberately incomplete problem representation in the form of a decision tree diagram, and another that was not given any representation aid. The findings revealed that the incomplete representation actually impaired performance because subjects tended to rely on it as a comprehensive and veridical representation of the problem, thereby failing to consider the important factors which had been deliberately omitted from the representation. Thus, being provided with an incomplete problem representation can actually lead to worse performance than having no representation at all. This can be attributed to the "out of sight, out of mind" phenomenon identified by Fischoff et al.

A final example of the strong tendency towards surface control comes from the domain of mathematics. Several researchers in this area (e.g. [19, 57, 99]) have observed "the attraction of surface structure over deep structure" [57, p. 77] exhibited by mathematics students. In cases where the mathematical symbols provide a faithful externalization of the underlying concepts being represented, this strategy can actually be adaptive (cf. [31]). However, the tendency for surface control can lead to problems when the mapping between surface and depth is not an isomorphic one [57]. An excellent example is provided by Dufour-Janvier et al. [19] who discuss the errors that children sometimes exhibit in using the number line as an aid to solving problems. This external representation consists of a horizontal line with the integers labelled and marked by tick marks at regularly spaced intervals along the line. Just as in the studies reviewed above, students have a tendency to equate this external representation and the mathematical concepts being represented. This strategy does not lead to trouble when students are learning about integers, but when the same external representation is used to reason about real numbers, several types of errors are regularly observed [19, p. 1 17]. For instance, children think that there are no other numbers, or at most one, between two whole numbers. Furthermore, students also find it difficult to place a number if they cannot associate it with the gradations already marked on the line. These examples clearly illustrate that when the mathematical concepts (deep structure) are not uniquely mapped onto salient perceptual features of the external representation (surface structure), the tendency to rely on perceptual processing can lead to significant and predictable errors.

Of course, there are several reasons why subjects in these various studies would fail to consider information not explicitly represented in the display made available to them. The most obvious, of course, is ignorance [22]. Perhaps the reason why subjects in these studies relied almost exclusively on the information in the displayed representation was that they did not have any other relevant knowledge to rely on. While this explanation may explain the errors made by students learning mathematics, it cannot account for all of the available data. The subjects in Hollnagel's [39] study were control engineers, not novices, and thus should have had sufficient domain knowledge to go beyond the information provided in the interface. More direct evidence rejecting the ignorance explanation comes from the final experiment reported by Fischoff et al. [22], where the subjects were technical experts instead of novices. This expertise manipulation did not affect the previously observed tendency. Both novices and experts failed to appreciate the material omitted from the diagrams with which they were presented.

3) Conclusions. Collectively, these studies provide strong empirical evidence indicating that people have a definite preference for lower levels of cognitive control, or surface control. The work of Klein [49] and Kirlik [48] shows the proficient level of performance that can result from this behavior. The other studies reviewed above reinforce the idea that people have a dis-

tinct tendency to engage in lower levels of cognitive control. Furthermore, these studies also illustrate the types of problems that are encountered if the interface does not support this strategy. The basic implication that can be derived from these findings is that interfaces should be designed to allow people to effectively meet the demands of the task by relying on lower levels of cognitive control.

C. Skill and Task Effects

So far, evidence for the power of lower levels of cognitive control and for people's preference for relying on those levels has been reviewed, leading to the recommendation that interfaces should be designed to support perceptual processing. At the same time, it is important to realize that just because information is presented in such a way that task demands could be, for example, satisfied through SBB alone, higher levels of cognitive control may nevertheless be activated. The reason for this is that the form in which information is presented does not directly determine which level of cognitive control will be activated (cf. [35, 79]). Several other factors are also important.

In particular, the level of cognitive control will also vary as a joint function of the level of skill of the operator and the level of complexity of task demands. Skill and task complexity are in fact duals, as Leplat [53] has pointed out, so it is impossible to talk about one without the other. The important point for the present discussion is that the psychological demands imposed by a given task depend on the level of skill of the operator [53, 68, 75, 941. As a result, the degree to which operators can effectively rely on lower levels of cognitive control is a function of their level of skill and experience. More experienced operators are able to deal with most task demands by relying on lower levels of cognitive control (cf. [18, 49, 68, 79, 89, 94]). In summary, the current demands of the task, the person's experience, and the form in which information is presented all combine to determine which level of cognitive control is activated.

Because operators may engage in higher levels of cognitive control (e.g., KBB) even if the interface is designed to encourage lower levels, merely supporting the lower levels is not sufficient. To be truly effective, an interface should also support higher levels of cognitive control. How can this be achieved? In order to determine the information requirements for each of the levels of cognitive control, it is necessary to understand how the different levels are related, and what the activities associated with each level are.

D. Interaction Between Levels

' While it is possible to describe each level independently, performance of a realistically complex task will usually require a simultaneous consideration of all three levels of cognitive control [88, 89]. However, the activities associated with each level are quite different from each other. For instance, the information required for on-line control of the current activity, and off-line planning considerations may not belong to the same time frame, nor to the same part of the problem space. 'ne information presented to the operator will have at least three distinct functions in the control of a complex work sequence. Information must be available for activation of skilled routines, control of the course of the routines, and monitoring the outcome of an activity [1081.

The important implication is that the type of computer support required to deal with these different classes of activities is not the same. A fiamework for interface design must take this into account.

E. Implications for Interface Design

The path of reasoning followed to this point is summarized in Figure 2. First, it has been argued that, lower levels of cognitive control tend to be executed more quickly, more effectively, and with less effort than higher levels. Second, converging empirical evidence argues that people have a definite preference for carrying out tasks by relying on lower levels of cognitive control, even when the interface is not designed to support this type of behavior [19, 22, 39, 57, 99, 100]. These two points suggest that information should be presented in a way that allows operators to effectively rely on lower levels of cognitive control. However, even if information is presented in such a way that a task can be accomplished using lower levels of cognitive control, higher levels may nonetheless be triggered because the level of cognitive control activated is determined not only by how information is presented but also by task demands and the operator's level of skill. This, and the fact that any reasonably complex task will require a complex interaction between all three levels of cognitive control, suggests that an interface should provide the appropriate support for all three levels. Therefore, the general goal that a framework for interface design should strive to achieve is to design interfaces in such a way as not to force cognitive control to a higher level than the demands of the task require, while at the same time providing the appropriate support for all three levels.

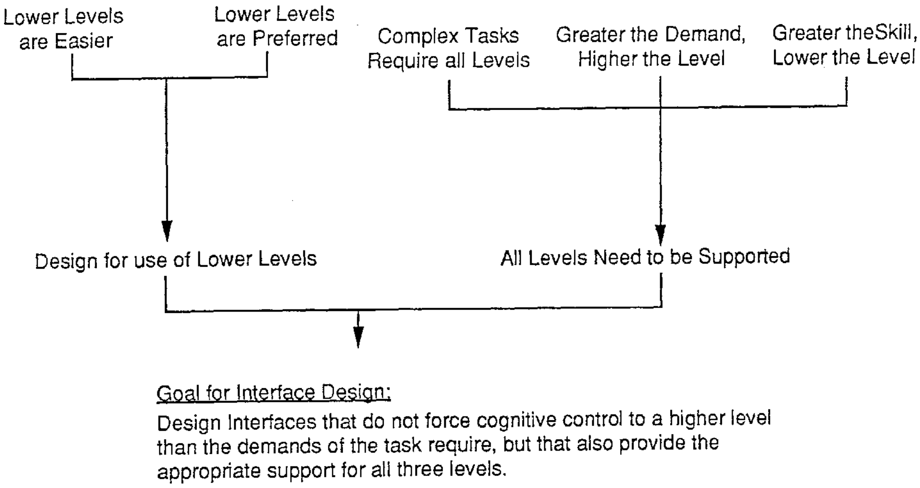

Figure 2. The deductive path leading to a basic goal for interface design.

The next step is to develop a set of prescriptive principles that will allow designers to develop interfaces that satisfy this goal. SRK provides a general indication of how to do this, since there are constraints on inducing each level of cognitive control. For instance, SBB can only be activated when information is presented in the form of time-space signals. RBB, on the other hand, is triggered by familiar perceptual forms (signs). And finally, KBB is activated by meaningful relational structures (symbols).

V. ECOLOGICAL INTERFACE DESIGN

The preceding theoretical development, combined with a morphological analysis of human-system interaction modes (cf. [108]), leads to the postulates of EID. The framework consists of three general principles, each corresponding to a specific level of cognitive control. ne intent is to develop a single design that will simultaneously support all three levels of cognitive control. In this section, each of the principles will be described and their theoretical significance within the context of process control will be discussed (again, see [107] for an application of the principles). Although many of the general ideas behind EID have been around for some time (see [7, 81, 83, 1 1 1] and especially [32, 33]), the specific theoretical formulation presented here is a new one.

A. The Principles

1. SBB - To support interaction via time-space signals, the operator should be able to act directly on the disl2lay. and the structure of the displayed information should be isomorphic to the part-whole structure of movements.

This principle attempts to structure the interface so as to take advantage of SBB. Because the operator cannot directly act on the plant components, the motor control patterns at the SBB level will only be concerned with the manipulation of objects displayed in the interface. The use of a mouse or a trackball is preferred to command languages for this task because it maintains the communication of spatial-temporal aspects of the perception-action loop intact. This is the familiar idea of control via direct manipulation [97].

The mapping or coupling between perception and action is also critical to SBB. Skilled perceptual-motor performance is characterized by integrated patterns of movements resulting in a very high capacity and speed in performance. A good example is musical skill [108]. As the musician's level of proficiency increases, movements are aggregated together into higher order chunks. Whereas the novice must control at the level of individual actions, skilled musicians can work at the level of complex sequences of actions. The key requirement for attaining this type of skill seems to be the mapping between the musical notation and the associated actions. Thus, experienced musicians are able to form higher order visual chunks of notes and then directly map these onto a concurrent chunking of movements. A similar situation can be found in the skilled use of an abacus [108].

This suggests that in order to facilitate sldlled perceptual-motor performance in complex work domains, a similar mapping should be built into the interface. Thus, the interface should be designed in such a way that the aggregation of elementary movements into more complex routines corresponds with a concurrent integration (i.e., chunking) of visual features into higher level cues for these routines. In other words, the structure of the displayed information should be isomorphic to the partwhole structure of movements. This can be accomplished by revealing higher level information as an aggregation of lower level information (e.g. through appropriate perceptual organization principles, see [23]). In this way, multiple levels are visible at the same time in the interface and the operator is free to guide his attention to the level of interest, depending upon his level of expertise and the current demands. Developing such a hierarchical visual structure should facilitate the acquisition of skill by encouraging the chunking process. At the same time, flexibility is maintained by not constraining people to attend to a specific level of description.

2. RBB - Provide a consistent ome-to-one mapping between the work domain constraints and the cues or signs provided by the interface,

This second principle attempts to support the RBB level. At this level, the display provides operators with signs that they use as cues for the selection of an appropriate action. The problem with conventional interfaces is that there is no consistent mapping between the perceptual cues that they provide and the constraints which govern the process' behavior. This leads to procedural traps [76]: novel situations where operators rely on their normal rule set, but without the usual success (see also [891).

A poignant example of a procedural trap can be found in the events that transpired at TMI (cf. [561). Operators had adopted pressurizer level as a cue for determining the total primary inventory. This heuristic worked most of the time, when the primary loop was in its usual state. The problem is that the operators were not aware of the boundary conditions under which the cue was valid. Thus, when the reactor coolant system (RCS) reached saturation conditions with resulting void formation, the pressurizer level was no longer a valid indicator of primary inventory. Yet, operators used the cue just as they normally would. As a result, they incorrectly inferred that there was a great deal of water in the pressurizer when in fact it was mainly full of steam.

EID attempts to overcome the difficulty associated with procedural traps by developing a unique and consistent mapping between the constraints that govern the behavior of the process, and the cues provided by the interface. This should reduce the frequency of errors because the cues for action, being based on fundamental process properties, will be uniquely defining of the underlying system state. This means that it is possible, in principle, for the operator to often effectively control the system by relying on perceptual cues rather than by having to resort to KBB. (For an excellent set of examples of this point within the contexts of ship navigation and algebra expressions, see Hutchins [40] and Goldstein [3 *1*], respectively).

There are two advantages to this strategy. The first is related to mental economy: the RBB level of cognitive control is less effortful than the KBB level. The second advantage of this design approach is that because there is a 1: I mapping between symbols and signs, the operator can exhibit what looks like KBB by merely relying on RBB. ne advantage of knowledge-based control is that, being based on fundamentals, its applicability is not restricted to specific conditions (e.g., frequently encountered scenarios) as RBB often tends to be. Therefore, the second principle of EID allows operators to take advantage of the cognitive economy of RBB while, at the same time, preserving the wide applicability of KBB.

Returning to the TMI example, the interface should have provided operators with diagnostic cues that would have allowed them to directly see that the RCS pressure had reached saturation conditions. In other words, rather than merely providing a heuristic cue like pressurizer level, the interface should have provided operators with perceptual information that reflects the fundamental constraints governing the process (in this case, the Rankine cycle heat engine). Interestingly, Beltracchi [4, 5] has developed an overview display for NPP's based on a plot of the Rankine cycle in temperature-entropy coordinates that is consistent with the EID philosophy.

It is encouraging to note that the utility of this principle of EID has received empirical support from a number of recent studies, which have shown that visual displays possessing emergent features that map onto goal-relevant constraints can lead to better performance than displays which do not provide such a mapping [10, 95].

More recently, Kaplan and Simon [45] have conducted a very interesting study of factors affecting how well people solve insight problems that provides indirect support for this design principle. They presented subjects with three versions of an insight problem which differed according to how perceptually salient the critical attribute for solving the problem was. The results revealed that insight was greatly facilitated if the critical attribute for solving the problem was displayed in a perceptually salient manner. Furthermore, Kaplan and Simon also observed that one of the key differences distinguishing fast from slow subjects was that the former tended to pay attention to the perceptual invariants in the display more often than the latter. But of course, paying attention to perceptual invariants will not be productive unless those invariants are meaningful. Thus, building interfaces where domain invariants are mapped isomorphically onto perceptual invariants takes advantage of people's propensity for perceptual processing so as to guide their attention to the meaningful attributes of the problem.

3. KBB - Represent the work domain in the form of an abstraction hierarchy to serve as an externalized mental model that will support knowledge-based problem solving.

This final principle attempts to provide the necessary support for KBB. This is essential to the success of an interface since KBB is usually an effortful and error prone activity. In part, the difficulty of successfully relying on KBB can be attributed to the complexity of high-tech systems. In these domains, problem solving takes place within the context of a complex causal network of relations. It is, therefore, very difficult for operators to ensure that all of the consequences of the action they select have been taken into account and evaluated [21, 77, 89].

The approach of EID to this problem is to reveal the problem space, in the form of the abstraction hierarchy, to the operator (see [107] for an example). The third principle therefore inherits all of the properties of the abstraction hierarchy described earlier: it provides a psychologically relevant domain representation that contains the information operators need to cope with unanticipated events. In addition, this design principle facilitates KBB by relieving operators of having to keep track of the complex causal network they are reasoning within. By making the abstraction hierarchy visible in the interface, the EID approach is providing the operator with a normative externalized mental model of the process that can support thought experiments and other planning activities.

There is empirical evidence to suggest that external representations of goal-relevant domain properties can improve problem solving performance. For example, Beveridge and Parkins [6] demonstrated that a visual representation of Duncker's [20] x-ray problem served as a facilitator for analogical problem solving. This particular study is important because Gick and Holyoak [29] had previously found that a diagram had not improved performance on the same problem. However, Beveridge and Parkins were able to show that the reason why Gick and Holyoak's visual representation did not help was because it did not represent the critical features of the x-ray problem in a perceptually salient manner. In contrast, their own diagram was specifically designed to represent the goal-relevant properties of the problem.

Kotovsky, Hayes, and Simon [51] also describe a study that provides empirical support for the design principle proposed above. They constructed various isomorphic versions of the Tower of Hanoi problem to see how presenting the problem in different forms affected performance. One of the findings to emerge from this study was that providing an external representation of the problem to be solved can improve performance by reducing subjects' memory load.

B. Limitations

Up to this point, a considerable body of literature providing both theoretical and empirical support for the EID framework has been presented. To provide a balanced perspective, however, the framework's limitations need to be addressed as well. The first three issues pertain to the use of the abstraction hierarchy. First**,** there are limitations imposed by designers' knowledge of the constraints governing the system. If those constraints are unknown, an abstraction hierarchy cannot be developed. Thus, the approach will only succeed to the extent that designers understand the system they are building.

Second, there is the question of robustness. Because data from sensors are inherently noisy [1 10] and therefore uncertain, and because the system model on which the redundant constraints are based is never known exactly [25], there will always be some deviation between expected normal behavior and the data observed by operators, even under normal operations. Empirical research is needed to determine how robust performance with an interface based on the abstraction hierarchy is with respect to these sources of uncertainty.

Third, there may also be limitations due to sensor technology. In some systems, there may be certain variables (particularly higher order functional information) that cannot be measured with existing sensors. In some cases, however, it may be possible to overcome this limitation by the use of analytical techniques (for an example, see the application of observer theory to derive the compensated level for a system with non-minimum phase dynamics in [*1 15]).*

The final question that needs to be addressed is that of generalizability. The ideas presented here were developed within the domain of process control, but the applicability and utility of the EID framework has yet to be systematically explored beyond this context. Nevertheless, it is important that the issue of generalization of these principles to other domains be addressed.

There are several reasons for being optimistic about the applicability of the EID framework to other work domains. Some of these are conceptual. The primary prerequisite for applying the framework is that the designer have a description of t,-Ie goal-relevant constraints governing the work domain. In principle, it is irrelevant what those constraints are, as long as they can be described in some way so that they can then be mapped onto perceptual features of the display. This second step of revealing these constraints is, in principle, limited only by the designer's imagination and the state of the art knowledge of how perception operates. There are also more pragmatic reasons for being optimistic about the framework's generalizability. Interfaces based on ideas very similar to those proposed here have been built for the domain of information retrieval in libraries [34, 70, 71, 72] and are currently being built for the domain of aviation [52]. The library system has already been evaluated and the response from users has been overwhelmingly positive [341.

To conclude, the EID framework is intended to apply to a wide variety of work domains where operators are required to cope with unanticipated events. While the anecdotal evidence from library and aviation domains just presented is encouraging, it does not provide a defensible basis for generalization. Until these principles are systematically applied to domains other than process control, and the efficacy of such applications rigorously evaluated, the generalizability of the EID framework remains an open question.

VI. WHAT IS THE CONTRIBUTION OF EID?

It is instructive to map the EID framework back onto the generic structure in Figure 1. One of the advantages of having formulated the problem of interface design in a generic form is that that structure can now be used to compare EID with other approaches to interface design. This can be done by taking each of the questions in Figure 1, looking at the answers EID proposes, and comparing these to the answers proposed by other researchers. This analysis will reveal what characteristics, if any, distinguish EID from other approaches to the interface design problem illustrated in Figure 1. The two questions will be addressed in reverse order.

A. Communicating the information to the operator

1) Direct manipulation interfaces. Clearly, the EII) approach has many similarities to direct manipulation interfaces (DNU). This should not be a surprise since the intention from the start was to extend the benefits of DMI to complex systems. But does EID go beyond existing theories of DMI (e.g., [41, 97]) in any substantive way?

Existing theories of DMI tend to be more descriptions than explanations or theories. They mainly emphasize the fact that DMI allow users to directly act on what they see in the display, as well as directly representing the objects of interest. What is missing, however, is an explanation in terms of human information processing capabilities. That is, if DMI are easier to use, then they must be allowing people to use processing mechanisms that are more efficient or more effective than those required to use more traditional interfaces. This is what the SRK framework provides EID that other theories of DMI do not possess: an explanation of the benefits of DMI in terms of general properties of human cognition as opposed to a description in terms of the interface technology (e.g., graphics, mouse, pop-up menus, etc.). This is a deeper explanation, rather than just a description, of the phenomenon being investigated.

Being based on the SRK framework, EID inherits a rich set of concepts, allowing one to make comparatively more precise statements. For instance, SRK points to the importance of describing the unique type of informational support that is required for problem solving activities (i.e., KBB). In contrast, none of the existing accounts of DMI make explicit reference to supporting problem solving through interface design. The fact that the SRK framework provides a more fine grained language also means that it is possible to go beyond describing DMI to actually laying out explicit principles for how to go about designing such interfaces. This seems to be the crucial difference between EID and existing theories of DMI.

2) Object displays. EID is also related to current research on object displays (e.g., [3, 10, I 1, 95]). The intent behind this body of work has been to apply existing knowledge from the area of visual perception to design displays that have higher order visual properties. Thus, not only will the elements be visible, but so will a more global relationship. By mapping the higher order perceptual relationships onto goal-relevant variables, certain types of tasks become much easier to perform. One does not have to integrate the individual elements to make a judgement about some higher order property of interest. Rather, one can perceive the higher order property directly.

Research on object displays is directly relevant to EID principle 2, which states that in order to support RBB, the perceptual cues (signs) in the interface should directly specify process constraints. Note that the principle does not state how this should be done, merely that it is beneficial to do so.

The work on object displays complements EID by providing specific recommendations on how to create salient perceptual cues that can then be used to reveal domain invariants.

3) Technology-driven display design. It is also appropriate to compare EID with research that is directed at how to best take advantage of computer graphics for building interfaces for complex technical systems. Much of this work (e.g., [46]) can best be described as building computerized mimic or schematic diagrams. The starting basis is the representation that technicians or operators already use, and efforts are made to embellish these representations with the powerful capabilities of dynamic, color graphics. The research emphasis is on making manipulations to the surface features of the interface (the manipulations being dictated by the capabilities that technology has made possible to date) and then investigating their effects on performance.

Certainly, there is much to be gained from this type of research. However, EID represents a different approach in that it starts off with knowledge, not about what technological capabilities are currently available, but about human capabilities and limitations, couched within the SRK framework. Thus, the EID approach is top-down whereas technology-based approaches are bottom-up. One of the disadvantages of a bottom-up approach is that new experiments are required every time there is a technological innovation. In contrast, the findings obtained from a top-down approach are not tied to any specific technological medium.

4) Summary. The differences uncovered in this cursory review suggest that E]ID has a unique contribution to make to the problem of interface design. While EID is certainly consistent with the idea of "making visible the invisible" that is such a strong part of DMI, existing accounts of DMI do not seem to capture the conceptual richness and resulting prescriptive capabilities of EID. It was also shown how current work on object displays fits into the general framework provided by EID. Finally, the contrast between the problem-driven approach of EID and the technology-driven approach that has dominated research on graphics-based displays was also discussed.

B. Representing the Complexity in the Domain,

With respect to domain representation (the first question in Figure 1), there is much less literature against which to compare EH). There are at least two reasons for this. First, the issue of domain representation is often simply not addressed. Second, many researchers base their computer interfaces on the information that is already provided to operators in mimic diagrams and schematics (e.g. [38, 46]). Typically, this involves displaying the state variables and not much else. However, there is no need to limit interfaces to merely providing the same information that operators already have, but in an electronic form. The flexibility of computer technology provides the capability for doing much more.

So far, the only alternative to the abstraction hierarchy that has been discussed is the default position of displaying the system state variables. The advantages of including the added higher level information represented in the abstraction hierarchy in an interface have already been mentioned. To reiterate, the abstraction hierarchy is a psychologically relevant representation that provides operators with an informational basis for coping with unanticipated events.

1) Operator function model. There is another formalism that can perhaps be thought of as an alternative to the abstraction hierarchy. Mitchell and Miller [63] have proposed a discrete control model, called operator function model (OFM), that can be used as a display design methodology. It should be noted, however, that the mapping between the abstraction hierarchy and OFM is not a simple one. The abstraction hierarchy is but one part of a comprehensive methodology for performing a cognitive work analysis [77]. In contrast, OFM is a self-contained analytical formalism which attempts to answer the following design questions: a) What data should be displayed? b) How should those data be organized into screens? c) How should context sensitivity be built into the display? d) How can information be presented at various levels of detail? In what follows, the abstraction hierarchy and OFM will be compared only in terms of the respective ways in which they can be used to detemiine what information to display in an interface. Note that the intent is only to compare OFM with the abstraction hierarchy, not all of EID, since OFM is silent on the question of the form in which information should be displayed.

OFM is intended as a normative model of operator behavior [63]. It has a heterarchical-hierarchical structure, with the top level of the hierarchy representing the control functions that the operator must perform. Each of these is mapped onto a set of subfunctions, which in turn are mapped onto tasks, and finally onto actions. The model also incorporates next-state transition functions that represent the meaningful ways in which one can move within the network structure. An important property of OFM is that these mappings are non-deterministic. That is, OFM does not lay out the single, idealized set of tasks and actions that are required to accomplish a given operator function, but instead identifies the set of tasks and actions which can be used (i.e., which are meaningful within the context of a given function). This display design methodology has been applied to a manufacturing system [59, 63, 64] and to a satellite ground control system [65].

How does this structure compare to that of the abstraction hierarchy? It is always difficult to compare frameworks that have been developed independently, each with their own unique language and theoretical suppositions. Fortunately, however, Miller [60] has performed an important service by developing a generic set of systems theoretic definitions of terms for human-machine systems work. These definitions can be adopted as a relatively neutral language for comparing the abstraction hierarchy and OFM. For the present purposes, the most important distinction made by Miller is that between a structural representation and a behavioral representation. A "structural representation is one in which the structures which define the system are defined directly in some set of objects" [60, p. 24]. A behavioral representation, on the other hand, is a representation of a dynamic system whose elements consist of system behaviors [60, p. 33]. This distinction is relevant to capturing the difference between the two formalisms. The abstraction hierarchy is a structural representation of the controlled system, whereas OFM is a behavioral representation of operator actions. While the difference may seem subtle, it does have important implications.

With its emphasis on operator action, OFM requires that operator decision making functions be explicitly represented. This means that, in contrast to the abstraction hierarchy, an OFM representation cannot support problem solving activities which are unpredictable, such as those associated with unanticipated disturbances in complex systems. This advantage of the abstraction hierarchy is most relevant in very complex work domains with many degrees of freedom where the operator must deal with situations that cannot be anticipated by the designer.

OFM's weak emphasis on providing support for unanticipated events can be understood if one examines its historical origins. The OFM formalism was adapted from a methodology developed by Miller [59, 61] for developing discrete control models (DCM) based on finite-state descriptions. The DCM methodology was intended to model "the set of discrete tasks by which the system configuration and mode of operation is established, and the procedures by which the team members' activities are coordinated" [61, p. 4]. This quotation reveals that the DCM was primarily directed at work domains whose demands are primarily procedural in nature. This emphasis is substantiated by the domains to which the DCM approach has been applied. The original application was to model three operators controlling an antiaircraft artillery system [59, 61], and a subsequent application was in modelling single operators performing a capture tracking task [62]. The practical relevance of unanticipated events is not nearly as great in these domains as it is in more complex systems, such as NPP'S. Thus, it is not surprising to find that neither the DCM nor OFM approaches explicitly address this class of problems.

This comparison suggests that the abstraction hierarchy and OFM have quite different properties because they are primarily directed at different types of problems. The abstraction hierarchy is intended as a work domain representation for complex systems where unanticipated events are the biggest threat to system safety, whereas OFM is directed at situations where operators are required to dynamically select the relevant subset of data from a very large pool to carry out predictable tasks. This point is frankly acknowledged by Mitchell and Saisi [65] who state: "Whereas Rasmussen primarily addresses decision-making in novel situations, ... the operator function model represents decision-making in the normal operator functions of monitoring, fine tuning, as well as predictable fault detection, diagnosis, and compensation" (p. 574). Therefore, the two models are complementary in that the abstraction hierarchy attempts to lay out the constraints in the work domain that are relevant for control, whereas OFM represents current system state as a function of the current decision making function being conducted[4]

2) Goal-means network. Woods and Hollnagel [1 14] also discuss a formalism, called goal-means network (GMN), that can be used as a domain representation for interface design. The GMN is actually a variant of the abstraction hierarchy, and while there are differences in detail between the two, the GMN shares almost all of the properties of the abstraction hierarchy that were listed above.

Woods and Hollnagel [114] discuss how the GMN can be used as a basis for evaluating the cognitive demands placed on the operator. In particular, they illustrate how the GMN can be used to collect and integrate data to help operators answer questions about system state, and to map out problem solving situations. For the present purposes, the GMN can be considered roughly equivalent to the abstraction hierarchy as a domain representation formalism.

3) Summary. Compared to the problem of how to display information, there is little work done on what information should be included in an interface. Two potential alternatives to the abstraction hierarchy were identified, GMN and OFM. The abstraction hierarchy and the GMN are similar in nature, providing a hierarchical representation of the functional structure of the work domain. In contrast, OFM provides a behavioral representation of operator actions that complements the structural representation of the work domain provided by the abstraction hierarchy. However, the abstraction hierarchy has unique properties that are particularly well suited to the settings it was meant to address (i.e., complex work domains with many de-

---

[4] Although this issue has not been explicitly researched, it is likely that the choice of model should be guided by the characteristics of the application. OFM will probably be more useful in highly proceduralized domains whereas the abstraction hierarchy will be more useful for work domains where unanticipated events are an important threat to system safety.

grees of freedom where operators need to cope with situations that cannot be anticipated by designers).

VII. EMPIRICAL EVALUATION

In this section, an experiment evaluating part of the EID framework is briefly described. The subject of the evaluation was principle 3, which deals with support for KBB or problem solving behavior. According to EID, to properly support KBB an interface should represent the work domain at various levels defined by the abstraction hierarchy. The experiment was conducted in the context of DURESS (DUal Reservoir System Simulation), a thermal-hydraulic process simulation which was designed to represent some of the properties that make human-machine systems complex (cf. [106]). Two interfaces for DURESS were developed, one based on the principles of EID and another based on a more traditional format (see [107] for a detailed description of the two interfaces). The latter, which was referred to as the Physical (P) interface, only displayed the settings of the system components and the goal variables. The former, which was referred to as the Physical/Functional (P+F) interface, contained all of the information in the P interface and higher order functional variables as well. These functional variables correspond to the higher levels of the abstraction hierarchy. Thus, whereas the P interface contained only a subset of the levels of the abstraction hierarchy for DURESS, the P+F interface represented the entire hierarchy. The purpose of the study was to determine how well these two interfaces supported problem solving in unfamiliar and unanticipated situations (i.e., KBB). A brief description of the experiment and some of the results follows (for more details, see [106]).

Theoretical experts and novices viewed dynamic event sequences showing the behavior of DURESS with the P and P+F interfaces. As mentioned earlier, the P+F interface represented all levels of the abstraction hierarchy whereas the P interface did not. There were three types of trials: Normal where the system was operating correctly, Fault where a single fault was introduced, and Random where the system's behavior did not obey physical laws. On each trial, subjects were asked to recall the final state of each of the process variables and to diagnose the system state.

In keeping with the objective of studying KBB, the experimental conditions were intentionally set up to evaluate how well the two interfaces support performance in novel situations and to avoid reliance on RBB (i.e., perceptual processing). Thus, the data were conducted under the following constraints. Subjects were not given any feedback during the entire experiment. They were only given one session to become familiar with each interface. Also, they did not receive any formal training on the operation of the system or on fault diagnosis. They were also not told what faults were going to appear, nor what the ratio of fault to normal to random trials was. In addition, the form and layout of the recall display bore no physical resemblance to either interface. Finally, subjects had to make a diagnosis after viewing the system for only 25 to 30 seconds. In sum, subjects were faced with unfamiliar events and were not given any external aids except the interface itself.

Several interesting findings were obtained. The P+F interface resulted in superior diagnosis performance compared to the P interface, primarily for experts. This result indicates that including all levels of the abstraction hierarchy can result in better support for knowledge-based problem solving than providing operators with physical state information alone, and that this result is due to a better match to the theoretical expert's mental model of the system. It was also found that diagnosis performance was significantly correlated with memory for functional variables; accurate diagnosis was associated with accurate memory for functional variables. In contrast, diagnosis performance and memory for physical variables were not significantly correlated. These results indicate that the higher-order functional information represented in the P+F interface is important for diagnosis, thereby justifying the argument for including higher levels of the abstraction hierarchy in an interface. The superiority of the P+F interface was also revealed in the memory task; memory for functional variables (i.e., those most critical to diagnosis) on meaningful trials (i.e., normal or fault) was better with the P+F than with the P interface. Furthermore, memory on random trials was significantly worse than on meaningful trials. This latter result indicates that the memory superiority of the P+F interface on meaningful trials cannot be attributed to differences in visual form between the two interfaces since the visual appearance of the P+F interface is identical on meaningful and on random trials. Thus, the superiority of the P+F over the P interface must, in part at least, be attributed to the added levels of abstraction in the P+F interface.

Collectively, these results are consistent with the following generalization: An interface based on an abstraction hierarchy representation of the work domain can provide more support for KBB than an interface based on physical variables alone because it results in a better match to the theoretical expert's veridical mental model.

VIII. SUMMARY

The framework described in this paper was motivated by a problem: how to design interfaces for complex work domains. The first step taken towards solving this problem was to determine what type of demands were associated with the control of complex systems. This analysis revealed that events which are unfamiliar and unanticipated pose the greatest threat to system safety. Thus, a viable approach to interface design for this class of systems must be able to support operators during unanticipated events. The next step taken was to formulate the generic structure of the interface design problem, the minimal set of questions to which any approach to interface design must provide answers. In the remainder of the paper, we developed a framework, EID, that attempts to provide useful answers to these questions. Both theoretical and empirical evidence was cited in support of the proposed framework. Furthermore, a review of a number of other approaches to interface design suggested that EID has a unique and useful contribution to make. Finally, an experiment providing some initial support for the framework was briefly described. As far as we know, this is the first study to compare an interface based on the abstraction hierarchy with any other type of interface [106]. The results suggest that the EID framework may have some value in achieving the goal that was posed at the beginning of this paper, i.e., to extend the benefits of DMI to the unique challenges posed by complex humanmachine systems.

Nevertheless, it is clear that many issues remain unanswered. There is one in particular that stands out. 'Me experimental results cited above indicate that an interface based on the principles of EID can lead to better performance for theoretical experts. It remains to be demonstrated whether this performance difference between interfaces will still hold with subjects who are not theoretical experts. Thus, the next step planned in this research is to give novice subjects extensive practice at controlling DLTRESS with either the P or P+F interface and see how skill acquisition, strategies, and performance on both normal and abnormal events vary as a function of interface. An experiment such as this would complement the one described above by focussing on issues related to the rule-based level of cognitive control.

There are also some ways in which the EID framework may be expanded. In its present form, the framework only concentrates on revealing the constraints inherent in the work domain in a form that is easy to perceive. However, there are other sources of constraint above and beyond those associated with the controlled system which could perhaps also be revealed in the interface in a manner similar to that described above. Examples include the constraints imposed by the particular control tasks required of the operators, the set of strategies that operators might adopt in performing those control tasks, the boundaries for safe operation defined by risk analyses, and regulatory policies and rules on system operation. Each of these layers of constraint needs to be taken into account by operators. It would be interesting to pursue the possibility of embedding and perhaps integrating these various layers of constraint in the interface through multiple layers of visual form. This idea, and the proposed experimental work described above, remain as topics for future research.

ACKNOWLEDGEMENTS

We would like to thank John Flach, Penny Sanderson, and especially Alex Kirlik for many influential discussions and helpful comments on earlier drafts, and Suzanne Faikus for help with the figures. Also, many thanks to the anonymous reviewers for providing thorough and thoughtful critiques of the original manuscript.

REFERENCES

[1] T. F. H. Allen and T. B. Starr, Hierarchy: Perspectives for Ecological ComplexiLy. Chicago: University of Chicago Press, 1982.

[2] W. R. Ashby, An Introduction to Cybernetics. London, Methuen: 1956.

[3] B. J. Barnett and C. D. Wickens, "Display proximity in multicue information integration: The benefit of boxes," Human Factors, vol. 30, pp. 15-24, 1988.

[4] L. Beltracchi, "Energy, mass, model-based displays, and memory recall," IEEE Transactions on Nuclear Science, vol. 36, pp. 1367-1382, 1989.

[5] L. Beltracchi, "A direct manipulation interface for water-based rankine cycle heat engines," IEEE Systems, Man, and Cybernetics, vol. SMC-17, pp. 478-487, 1987.

[6] M. Beveridge and E. Parkins, "Visual representation in analogical problem solving," Memoiy and CoL-nition, vol. 15, pp. 230-237, 1987.

[7] B. Brehmer, "Man as an operator of systems," in H. E. Peterson and W. Schneider, Eds., Human-Computer Communications in Health Care, pp. 7-15. New York: Elsevier, 1986.

[8] E. Brunswik, The Conceptual Framework of Psychology. Chicago,]]L: University of Chicago Press, 1952.

[9] E. Brunswik, Percel2tion and the Rel2resentative Design of Experiments, 2nd ed. Berkeley, CA: University of California Press, 1956.

[10] M. A. Buttigieg and P. M. Sanderson, "Emergent features in visual display design for two types of failure detection tasks," Manuscript submitted for publication, 1989.

[11] C. M. Carswell and C. D. Wickens, "Information integration and the object display," Ergonomics, vol. 30, pp. 511-527, 1987.

[12] W. G. Chase and M. T. H. Chi, "Cognitive skill: Implications for spatial skill in large-scale environments," in J. H. Harvey, Ed. Cognition. Social Behavior. and the Environment, pp. 1 1 1- 136. Hillsdale, NJ: Erlbaum, 198 1.

[13] M.T.H.Chi, P.J.Feltovich,and R.Glaser,"Categorization and representation of physics problems by experts and novices," Cognitive Science, vol. 5, pp. 121-152, 1981.

[141 R. C. Conant and W. R. Ashby, "Every good regulator of a system must be a model of that system," International Journal of Systems Science, vol. 1, pp. 89-97, 1970.

[15] R. Davis, "Diagnostic reasoning based on structure and behavior," Artificial Intelli2ence, vol. 24, pp. 347-410, 1984.

[16] A. D. de Groot, "Perception and memory versus thought: Some old ideas and recent findings," in B. Kleinmuntz, Ed. Problem Solving: Research, Method. and Theorv I PP. 19-50. New York: Wiley, 1966.

[17] A. D. de Groot, Thought and Choice in Chess. The Hague: Mouton, 1965 (English translation of original Dutch version published in 1946).

[18] H. L. Dreyfus and S. E. Dreyfus, Mind Over Machine: The Power of HUMAN Intuition and Expertise in the Era of the Computer. Oxford: Basil Blackwell, 1986.

[19] B. Dufour-Janvier, N. Bednarz, and M. Belanger, "Pedagogical considerations concerning the problem of representation," in C. Janvier, Ed. Problems of Representation in the Teaching and Learning of Mathematics, pp. 109-122. Hillsdale, NJ: Erlbaum, 1987.

[20] K. Duncker, "On problem solving," Paychological Monographs, vol. 58, Whole No. 270, 1945.

[21] D. Dörner, "On the difficulties people have in dealing with complexity," in J. Rasmussen, K. Duncan, and J. Leplat, Eds. New Technology and Human Error, pp. 97-109. Chichester, UK: Wiley, 1987.

[22] B. Fischoff, P. Slovic, and S. Lichtenstein, "Fault trees: Sensitivity of estimated failure probabilities to problem representation," Journal of Experimental Psychology: Human Percel2tion and Performance, vol. 4, pp. 330-344, 1978.

[23] J. M. Flach and K. J. Vicente, "Complexity, difficulty, direct manipulation and direct perception," EPRL-89-03, Department of Mechanical and Industrial Engineering, University of Illinois at Urbana-Champaign, Urbana, IL, 1989.

[24] B. A. Francis and W. M. Wonham, "The internal model principle of control theory," Automatica, vol. 12, pp. 457-465, 1975.

[25] P. M. Frank, "Fault diagnosis in dynamic systems using analytical and knowledge-based redundancy - A survey and some new results," AUIQmWiga, vol. 26, pp. 459-474, 1990.

[26] N. H. Frijda and A. D. de Groot, Otto Selz: His ContributiiQ@P @chholoL-y. The Hague: Mouton, 198 1.

[27] J. J. Gibson, The Ecological approach to Visual Percel2fion. Boston: Houghton-Mifflin, 1979.

[28] J. J. Gibson, The Senses Considered a& Perceptual-Systems. Boston: Houghton-Mifflin, 1966.

[29] M. L. Gick and K. J. Holyoak, "Schema induction and analogical transfer. Cognitive Psychology, vol. 15, pp. 1-28, 1983.

[30] R. Glaser and M. T. H. Chi, M. T. H., "Overview," in M. T. H. Chi, R. Glaser, and M. J. Farr, Eds. The Nature of Exl2ertise, pp. xv-xxviii. Hillsdale, NJ: Erlbaum, 1988.

[31] M. Goldstein, "Display aspects of algebra," Psychological Rel2orts, vol. 24, pp. 937-938, 1969.

[32] L. P. Goodstein, "Discriminative display support for process operators," in J. Rasmussen and W. B. Rouse, Eds. Human Detection and diagnosis of System Failures, pp. 433-449. New York: Plenum, 198 1.

[33] L. P. Goodstein, "Operator communication in modem process plants," in Proceedings of the IEE Conference on Displays, pp. 149-153. London: Institution of Electrical Engineers, 1971.

[34] L. P. Goodstein and A. M. Pejtersen, "The book house: System functionality and evaluation," Riso-M-2793, Department of Information Technology, Riso National Laboratory, Roskilde, Denmark, 1989.

[35] R. M. Hamm, "Moment-by-moment variation in experts'analytic and intuitive cognitive activity," IEEE Transactions on Systems, Man. and Cybernetics, vol. SMC-18, pp. 757-776, 1988.

[36] K. R. Hammond and D. A. Summers, "Cognitive control," Psychological Review, vol. 79, pp. 58-67, 1972.

[37] K. R. Hammond, R. M. Hamm, J. Grassia, and T. Pearson, "Direct comparison of intuitive and analytical cognition in expert judgement," IEEE Transactions on Systems, Man, and Cybernetics, vol. SMC-17, pp. 753-770,1987.

[381 J. D. Hollan, E. L. Hutchins, T. P. McCandless, M. Rosenstein, and L. Weitzman, "Graphic interfaces for simulation," in W. B. Rouse, Ed. Advances in Man-Machine Systems Research, vol, 3, pp. 129-163. Greenwich, CF: JAI Press, 1987.

[39] E. Hollnagel, "Report from the third NKA/KRU-experiment: The performance of control engineers in the surveillance of a complex process," NKA/YRU-P2(81)36, Electronics Department, Riso National Laboratory, Roskilde, Denmark, 1981.

[40] E. Hutchins, "The technology of team navigation," in J. Galegher, R. E. Kraut, and C. Egido, Eds. Intellectual Teamwork: Social Foundations of Cooperative Work, pp. 191-220. Hillsdale, NJ: Erlbaum, 1990.

[41] E. L. Hutchins, J. D. Hollan, and D. A. Norman, "Direct manipulation interfaces," in D. A. Norman and S. W. Draper, Eds. User Centered System Design: New Persl2ectives on Human-Computer Interaction, pp. 87-124. Hillsdale, NJ: Erlbaum, 1986.

[42] International Electrotechnical Conunission, "Design for control rooms of nuclear power plants," International Standard IEC 964, International Electrotechnical Commission, Geneva, Switzerland, 1989.

[43] J. Itoh, S. Yoshimura, T. Ohtsuka, and F. Masuda, F., "Cognitive task analysis of nuclear power plant operators for man-machine interface design," in Proceedings of the ANS Tol2ical Meeting on Advances in Human Factors- Research on Man-Computer Interactions: Nuclear and Beyond, pp. 96-102. La Grange Park, IL: American Nuclear Society, 1990.

[44] A. Jensen, J. Rasmussen, and P. Timmerman, "Analysis of failure data for electronic equipment at Riso," Riso Report No. 38, Danish Atomic Energy Commission, Research Establishment Riso, Roskilde, Denmark, 1963.

[45] C. A. Kaplan and H. A. Simon, "In search of insight," CoLnitive Paycholo2y, vol. 22, pp. 374-419, 1990.

[46] D. Kieras, "Diagrammatic displays for engineered systems: Effects on human performance in interacting with malfunctioning systems," Tech. Rep. No. 29, Technical Communication Program, University of Michigan, Ann Arbor, MI, 1988.

[47] R. G. Kincade and J. Anderson, "Human factors guide for nuclear power plant control room development," EPRI NP-3659, Electric Power Research Institute, Palo Alto, CA, 1984.

[48] A. C. Kirlik, "The organization of perception and action in complex control skills," Unpublished Dissertation, The Ohio State University, Department of Industrial and Systems Engineering, Columbus, OH, 1989.

[49] G. A. Klein, "Recognition-primed decisions," in W. B. Rouse, Ed. Advances in Man-Machine Systems Research, vol, 5, pp. 47-92. Greenwich, CT: JAI Press, 1989.

[50] R. E. Korf, "Planning as search: A quantitative approach," Artificial Intelligence, vol. 33, pp. 65-88, 1987.

[51] K. Kotovsky, J. R. Hayes, and H. A. Simon, "Why are some problems hard? Evidence from Tower of Hanoi," Cognitive Psychology, vol. 17, pp. 248-294, 1985.

[52] C. F. Layton, P. J. Smith, and C. E. McCoy, "Graphical interfaces for cooperative planning systems," in Proceedings -Of the 1990 IEEE International Conference on Systems, Man, and Cybernetics, pp. 508-51 1. Los Angeles, CA: IEEE, 1990.

[53] J. Leplat, "Task complexity in work situations," in L. P. Goodstein, H. B. Andersen, and S. E. Olsen, Eds. Tasks. Errors and Mental models: A Festschrift to Celebrate the 60th Birthday of Professor Jens Rasmussen, pp. 105-115. London: Taylor and Francis, 1988.

[54] M. Lind, "Representations and abstractions for interface design using multilevel flow modelling," in G. Weir and J. Alty, Eds. Human-Coml2uter Interaction and Coml2lex Systems. London: Academic Press, in press.

[55] J. J. Lipsett, R. A. Olmstead, and J. E. S. Stevens, "Balancing the roles of humans and machines in power plant control," AECL-9955, Atomic Energy of Canada Limited, Chalk River Nuclear Laboratories, Chalk River, Canada, 1989.

[56]T. B. Malone, M. Kirkpatrick, K. Mallory, D. Eike, J. H. Johnson, and R. W. Walker, "Human factors evaluation of control room design and operator performance at Three Mile Island-2," NUREG/CR-1270, USNRC, Washington, DC, 1980.

[57] J. H. Mason, "What do symbols represent?," in C. Janvier, Ed. Problems of Reprsentation in the Teaching and Learning of Mathematics, pp. 73-8 1. Hillsdale, NJ: Erlbaum, 1987.

[58] M. D. Mesarovic, D. Macko, and Y. Takahara, Theory of Hierarchical. Multilevel, Systems. New York: Academic Press, 1970.

[59] R. A. Miller, "A systems approach to modelling discrete control performance," in W. B. Rouse, Ed. Advances in Man-Machine Systems Research, vol, 2, pp. 177-248. Greenwich, CT: JAI Press, 1985.

[60] R. A. Miller, "Formal analytical structure for manned systems analysis," Final Report, RF Project 762598n 13637, Department of Industrial and Systems Engineering, The Ohio State University, Columbus, OH, 1982.

[61] R. A. Miller, "Identification of finite state models of human operators," Final Report, AFOSR Grant #77-3152, Department of Industrial and Systems Engineering, The Ohio State University, Columbus, OH, 1979.

[62] R. A. Miller, R. J. Jagacinski, R. B. Nalavade, and W. W. Johnson, "A finite-state description of coordination in a two-handed target acquisition task," IEEE Transactions on Systems, Man, and Cybernetics, vol. SMC-12, pp. 529-538, 1982.

[63] C. M. Mitchell and R. A. Miller, "A discrete control model of operator function: A methodology for information display design," IEEE Transactions on Systems, Man, and Cybernetics, vol. SMC-16, pp. 343-357, 1986.

[64] C. M. Mitchell and R. A. Miller, "Design strategies for computer-based information displays in real-time control systems," Human Factors, vol. 25, pp. 353-369, 1983.

[65] C. M. Mitchell and D. L. Saisi, "Use of model-based qualitative icons and adaptive windows in workstations for supervisory control systems," IEEE Transactions on Systems, Man. and Cybernetics, vol. SMC- 17, pp. 573-593, 1987.

[66] J. R. Newman, "Extension of human capability through information processing and display systems," SP-2560, System Development Corporation, Santa Monica, CA, 1966.

[67] D. A. Norman, "Categorization of action slips," Psychological Review, vol. 88, pp. 1-15, 1981.

[68] S. E. Olsen and J. Rasmussen, "The reflective expert and the prenovice: Notes on skill-, rule-, and knowledge-based performance in the setting of instruction and training," in L. Bainbridge and S. A. R. Quintanilla, Eds. Developing Skills With Information TechnoloLry, pp. 9-33. Chichester: Wiley, 1989.

[69] H. H. Pattee, "The nature of hierarchical controls in living matter," in R. Rosen, Ed. Foundations of Mathematical BioloLy. Vol. 1 - Subcellular Svstems, pp. 1-21. New York: Academic Press, 1972.

[70] A. M. Pejtersen, "The book house: Modelling users' needs and search strategies as a basis for system design," Riso-M-2794, Department of Information Technology, Riso National Laboratory, Roskilde, Denmark, 1989.

[71] A. M. Pejtersen, "Search strategies and database design for information retrieval from libraries," in L. P. Goodstein, H. B. Andersen, and S. E. Olsen, Eds. Tasks, Errors, and Mental Models: A Festschrift to Celebrate the 60th Birthday-Qf Professor Jens Rasmussen, pp. 171-190. London: Taylor and Francis, 1988.

[72] A. M. Pejtersen and L. P. Goodstein, "Beyond the desk top metaphor: Information retrieval with an icon-based interface," in M. J. Tauber and P. Gomy, Eds. Visualisation in HumanComl2uter Interaction. Berlin: Springer-Verlag, in press.

[73] C. Perrow, Normal Accidents: Living With High-Risk Technologies. New York: Basic Books, 1984.

[74] M. Polanyi, Personal Knowledge : Towards a Post-critical Philosophy. Chicago: University of Chicago Press, 1958.

[75] J. Rasmussen, "Mental models and the control of action in complex environments," in D. Ackermann and M. J. Tauber, Eds. Mental Models and Human-Coml2uter Interaction 1, pp. 41-69. Amsterdam: North-Holland, 1990.

[76] J. Rasmussen, "Cognitive control and human error mechanisms," in J. Rasmussen, K. Duncan, and J. Leplat, Eds. New Technology and Human Error, pp. 53-61. New York: Wiley, 1987.

[77] J. Rasmussen, Information Processing and Human-Machine Interaction: An Approach to Cognitive Engineering. New York: North Holland, 1986.

[78] J. Rasmussen, "The role of hierarchical knowledge representation in decision making and system management," IEEE Transaction on Systems, Man, and Cybernetics, vol. SMC-15, pp. 234-243, 1985.

[79] J. Rasmussen, "Skills rules, knowledge: Signals, signs, and symbols and other distinctions in human performance models," IEEE Transactions on Svstems. Man. and Cybernetics, vol. SMC-13, pp. 257-267, 1983.

[80] J. Rasmussen, "On the structure of knowledge - A morphology of mental models in a man-machine system context," Riso-M-2192, Riso National Laboratory, Electronics Department, Roskilde, Denmark, 1979.

[81] J. Rasmussen, "The human data processor as a system component: Bits and pieces of a model", Riso-M-1722, Danish Atomic Energy Commission, Roskilde, Denmark, 1974.

[82] J. Rasmussen, "Man-machine communication in the light of accident records," S- 1-69, Danish Atomic Energy Commission, Research Establishment Riso, Roskilde, Denmark, 1969.

[83] J. Rasmussen, "On the communication between operators and instrumentation in automatic process control plants," Riso-M-686, Danish Atomic Energy Commission, Research Establishment Riso, Roskilde, Denmark, 1968.

[84] J. Rasmussen, "On the reliability of process plants and instrumentation systems," Riso-M- 706, Danish Atomic Energy Commission, Research Establishment Riso, Roskilde, Denmark, 1968.

[85] J. Rasmussen, "Characteristics of process operator, automatic equipment and designer in plant automation," Riso-M-808, Danish Atomic Energy Commission, Research Establishment Riso, Roskilde, Denmark, 1968.

[86] J. Rasmussen and M. Lind, "Coping with complexity," Riso-M-2293, Electronics Department, Riso Nationai Laboratory, Roskilde, Denmark, 198 1.

[87] J. Rasmussen, and P. Timmerman, "Safety and reliability of reactor instrumentation with redundant instrument channels," Riso Report No. 34, Danish Atomic Energy Commission, Research Establishment Riso, Roskilde, Denmark, 1962.

[88] J. Rasmussen and K. J. Vicente, "Coping with human error through system design: Implications for ecological interface design," International Journal of Man-Machine Studies, vol. 31, pp. 517-534.

[89] J. Reason, Human Error. Cambridge, Cambridge University Press, 1990.

[90] J. Reason, "Framework models of human performance and error: A consumer guide," in L. P. Goodstein, H. B. Andersen, and S. E. Olsen, Eds. Tasks, Errors, and Mental Models: A Festschrift to Celebrate the 60th Birthday of Professor Jens Rasmussen, pp. 35-49. London: Taylor and Francis, 1988.

[91] J. Reason, "Modelling the basic error tendencies of human operator," Reliability Engineering and System Safety, vol. 22, pp. 137-153, 1988.

[921 W. B. Rouse, "Models of human problem solving: Detection, diagnosis, and compensation for system failures," Automatica, vol. 19, pp. 613-625, 1983.

[93] P. M. Sanderson, "Knowledge acquisition and fault diagnosis: Experiments with PLAULT," IEEE Transaction on Systems, Man..and Cybernetics, vol. SMC-20, pp. 225242, 1990.

[94] P. M. Sanderson, and K. Harwood, "The skills, rules and knowledge classification: A discussion of its emergence and nature," in L. P. Goodstein, H. B. Andersen, and S. E. Olsen, Eds. Tasks, Errors. and Mental Models: A Festschrift to Celebrate the 60th Birthdav of Professor Jens Rasmussen, pp. 21-34. London: Taylor and Francis, 1988.

[95] P. M. Sanderson, J. M. Flach, M. A. Buttigieg, and E. J. Casey, "Object displays do not always support better integrated task performance," Human Factors, vol. 31, pp. 183-198, 1989.

[96] 0. Selz, Zur Psychologie des Produktiven Denkens und des Irrtums. Bonn: Freidrich Cohen, 1922.

[97] B. Shneiderman, "Direct manipulation: A step beyond programming languages," EEEE Computer, vol. 16, no. 8, pp. 57-69, 1983.

[98] H. A. Simon, The Sciences of the Artificial (2nd Ed.). Cambridge, MA: MIT Press, 1981.

[99] R. R. Skemp, "Communicating mathematics: Surface structures and deep structures," Visible Language, vol. 16, pp. 281-288, 1982.

[100] G. F. Smith, "Representational effects on the solving of an unstructured decision problem. IEEE Transactions on Systems, Man. and Cybernetics, vol. SMC-19, pp. 1083-1090, 1989.

[101] H. G. Stassen "On the modeling of manual control tasks," in G. R. McMillan, D. Beevis, E. Salas, M. H. Strub, R. Sutton, and L. V. Breda, Eds. Application of Human Performance Models to System Design, pp. 107-122. New York: Plenum, 1989.

[102] USNRC, "Guidelines for control room design reviews," NUREG-0700, United States Nuclear Regulatory Commission, Washington, DC, 198 1.

[103] USNRC, "Clarification of TMI action plan requirements," NUREG-0737, United States Nuclear Regulatory Commission, Washington, DC, 1980.

[104] USNRC, "TMI-2 lessons learned task force final report," NUREG-0585, United States Nuclear Regulatory Commission, Washington, DC, 1979.

[105] I. Vessey, "Expertise in debugging computer programs: A process analysis," International Journal of Man-Machine Studies, vol. 23, pp. 459-494, 1985.

[106] K. J. Vicente, "Supporting knowledge-based behavior through ecological interface design," Unpublished dissertation, University of Illinois at Urbana-Champaign, Urbana, IL, 1991.

[107] K. J. Vicente and J. Rasmussen, "The ecology of human-machine systems 11: Mediating 'direct perception' in complex work domains," Ecological Psycholo2y, vol. 2, pp. 207250, 1990.

[108] K. J. Vicente and J. Rasmussen, "A theoretical framework for ecological interface design," Riso-M-2736, Department of Information Technology, Riso National Laboratory, Roskilde, Denmark, 1988.

[109] G. R. S. Weir, "Varieties of dialogue in man-machine systems," in Proceedings of the 2nd European Meeting on Cognitive Science Approaches to Process Control, pp. 293-302. Siena, Italy: CEC-JRC Ispra, 1989.

[110]D. D. Woods, "Coping with complexity: The psychology of human behaviour in complex systems," in L. P. Goodstein, H. B. Andersen, and S. E. Olsen, Eds. Tasks, Errors, and Mental Models: A Festschrift to Celebrate the 60th Birthday of Professor Jens Rasmussen, pp. 128-148. London: Taylor and Francis, 1988.

[111] D. D. Woods, "Knowledge based development of graphic display systems. In Proceedings of the Human Factors Society 29th Annual Meeting. pp. 325-329. Santa Monica, CA: HFS, 1985.

[112] D. D. Woods, "Visual momentum: A concept to improve the cognitive coupling of person and computer," International Journal of Man-Machine Studies, vol. 21, pp. 229-244, 1984.

[113] D. D. Woods, "The cognitive engineering of problem representations," in J. Alty and G. Weir, Eds. Human-Computer Interaction and Complex Systems. London: Academic Press, in press.

[114] D. D. Woods and E. Hollnagel, "Mapping cognitive demands in complex problem-solving worlds," International Journal of Man-Machine Studies, vol. 26, pp. 257-275, 1987.

[115] D. D. Woods and E. M. Roth, "Aiding human performance U: From cognitive analysis to support systems," Le Travail Humain, vol. 51, pp. 139-172, 1988.

[116] D. D. Woods, J. F. O'Brien, and L. F. Hanes, "Human factors challenges in process control: The case of nuclear power plants," in G. Salvendy, Ed. Handbook of Human Factors, pp. 1724-1770. New York: Wiley, 1987.

BIOGRAPHIES

Kim J. Vicente received a B.A.Sc. (1985) in industrial engineering from the University of Toronto, a M.S. (1987) in industrial engineering and operations research from the Virginia Polytechnic Institute and State University, and a Ph.D. (1991) in mechanical engineering from the University of Illinois at Urbana-Champaign. During 1987-88, he spent one year as a visiting scientist in the Section for Informatics and Cognitive Science of the Riso National Laboratory in Roskilde, Denmark. He is currently an assistant professor in the Center for Human-Machine Systems Research, School of Industrial and Systems Engineering at the Georgia Institute of Technology. Kim is interested in interface design for complex human-machine systems, expertise, and more broadly, in the design and analysis of complex work environments.

Jens Rasmussen received a M.S. (1950) in electronic engineering from the Technical University of Copenhagen. In 1956 he joined the Atomic Energy Commission's Research Establishment Riso, now Riso National Laboratory. From 1962-1987 he was head of the Electronics Department. In 1962, he was elected as a member of the Danish Academy of Technical Sciences; in 1979-1981 he was chairman of a group of experts on human error analysis under the Committee on Safety of Nuclear Installations of OECD; and during 1981-1983, he was a member of the NATO Special Program Panel on Human Factors. He was visiting professor during 1982-83 at the Center for Man-Machine Systems Research, School of Industrial and Systems Engineering, Georgia Institute of Technology. From 1985-1987, he was a member of an expert panel on "Human factors regulatory research needs in nuclear power" under the National Research Council, National Academy of Sciences in Washington, DC. In 1987, he received the Distinguished Foreign Colleague Award from the Human Factors Society. He is currently a professor of cognitive engineering at Riso National Laboratory and the Technical University of Copenhagen. Since 1962 he has conducted research on reliability and risk analysis, human reliability, and human-machine interaction, and he has numerous publications in these areas.