Reprinted from

acta
psychologica

Acta Psychologica 91 (1996) 221–244

Supporting evaluation in design

Nathalie Bonnardel [a,*], Tamara Sumner [b]

[a] CREPCO (Centre de Recherche en Psychologie Cognitive) – URA CNRS 182, Université de Provence, 13621 Aix en Provence, France
[b] Department of Computer Science and Institute of Cognitive Science, University of Colorado, Boulder, CO 80309, USA

N·H

ELSEVIER

Acta Psychologica 91 (1996) 221–244

acta
psychologica

Supporting evaluation in design

Nathalie Bonnardel [a,*], Tamara Sumner [b]

[a] CREPCO (Centre de Recherche en Psychologie Cognitive) – URA CNRS 182, Université de Provence, 13621 Aix en Provence, France
[b] Department of Computer Science and Institute of Cognitive Science, University of Colorado, Boulder, CO 80309, USA

Abstract

Design problem-solving requires designers to be creative and to express evaluative judgments. Designers propose successive partial solutions and evaluate these solutions with respect to various criteria and constraints. Evaluation plays a major role in design because each successive evaluation step guides the course of design activity. However, evaluation of design solutions is difficult for both experienced and inexperienced designers because: (1) in complex domains, no single person can know all the relevant criteria and constraints, and (2) design solutions must be evaluated from multiple, and sometimes conflicting, perspectives. Domain-oriented design environments have been proposed as computational tools supporting designers to construct and evaluate design solutions. Critiquing systems embedded in these environments support evaluation activities by analysing design solutions for compliance with criteria and constraints encoded in the systems' knowledge-base. To investigate the impact of such systems, we have designed, built, and evaluated a domain-oriented design environment for a specific area: phone-based interface design. Professional designers were observed using the design environment to solve a complex design task. Analyses of these design sessions enabled us to identify reactions common to all designers, as well as reactions depending on the designers' level of domain experience.

PsycINFO classification: 2340, 4010, 4100, 4120

Keywords: Design; Problem-solving; Evaluation; Cooperative problem-solving systems; Critiquing systems

1. Introduction

In today's workforce, there are many skilled professionals engaged in design activities (e.g., in areas such as software design, mechanical design, or architectural design). Projections on the workforce of the next century indicate that skilled designers will

* Corresponding author. E-mail: nathb@romarin.univ-aix.fr, Fax: +33 42205905.

become increasingly prevalent in the years ahead (Quinn, 1992). Thus, there are very pragmatic reasons why it is important that the tools used by these designers effectively support their design activities and enable them to create better designs more efficiently. Towards this end, numerous research efforts are focused on creating various types of knowledge-based design support environments (see, for example, Eisenberg and Fischer, 1994; Fischer, 1994; Guindon, 1992). *However, a major challenge is to create knowledge-based design tools that are cognitively ergonomic; i.e., tools that effectively support designers' mental activities without hindering or interfering with their creative processes.*

To meet this challenge, we propose an iterative system development cycle where analyses of designers' cognitive processes both precede and follow system building. An understanding of the designers' traditional activities (i.e., without the design support system) can be used to guide system design in order to create a system better adapted to the designers' cognitive processes. The understanding of the designers' activities should also be used to inform system evaluation. Indeed, the evaluation of complex systems is confounded by the fact that their use substantially changes the traditional task (see, for example, Gould, 1980; Pavard, 1985; Whitefield, 1986); therefore, system evaluation cannot be limited to an assessment of the compliance of the systems' interfaces with general usability criteria or guidelines. Only an analysis of the designers' activities can be used as a baseline for assessing how the system has changed the task and whether the changes are for the better (Burkhart et al., 1994).

We illustrate this approach in a specific context: creating a knowledge-based design environment to support the design of voice dialog applications. Voice dialog applications are software systems with phone-based interfaces. During this project, we analysed traditional design activities and, in accordance to the obtained results, created a Voice Dialog Design Environment (VDDE). The VDDE system contains embedded, knowledge-based computational critics that support evaluation activities by automatically analysing partial design solutions for compliance with different user interface guidelines. The system interjects into the design activity by notifying the designer when a potential problem is detected in the design solution.

To understand the impact of VDDE's critiquing system on design activities, a study was performed with four professional voice dialog designers employed by a regional phone company. We asked these designers to use the Voice Dialog Design Environment to perform a design task corresponding to a real project, then we analysed comparatively these design sessions in order to identify reactions common to all designers, as well as specific reactions dependent on the designers' level of experience. We believe such studies are of interest both from a practical point of view, by suggesting how to build better design support systems, and from a theoretical point of view, by contributing to our understanding of design processes. Here, we have chosen to study professional designers in complex real-world design situations, rather than controlled laboratory situations, to better understand the impact of the design environment in a naturalistic setting. Despite the evident empirical limitations of such an exploratory study, we believe it allows the identification of patterns of system influence in designers' activities that are of interest to both design theorists and researchers concerned with the design of complex cognitive tools.

Before presenting this study, we first characterise design problem-solving and the aspects of it our design tools are trying to support (Section 2); and we describe the critiquing systems and the VDDE system in particular (Section 3). The study is then presented (Section 4), including: the underlying hypotheses, the experimental situation, the data analysis method, and the results obtained. We finally discuss the implication of these results for evaluation activities and knowledge-based design support tools (Section 5).

2. The role of evaluation in design

Contrary to traditional problems studied in Psychology, such as the 'Towers of Hanoi,' design problems are both 'ill-defined' and 'open-ended'. Design problems are considered *ill-defined* because designers have, initially, only an incomplete and imprecise mental representation of the design goals or specifications (Eastman, 1969; Reitman, 1965; Simon, 1981). Design problems are considered to be *open-ended* because there is usually no single correct solution for a given problem, but instead a variety of potential solutions (Fustier, 1989).

These characteristics lead to design processes involving an iterative dialectic between problem-framing and problem-solving (Schön, 1983). During problem-framing, designers refine design goals and specifications, and thus refine their mental representation of the problem. During problem-solving, designers elaborate solutions and evaluate these solutions with respect to various criteria and constraints. The results of these evaluation steps feed back into problem-framing and lead designers to focus their attention on specific features of the current solution which guides further development or modification of these features (Bonnardel, 1989, Bonnardel, 1992). Thus, the final solution to a design problem is obtained progressively, through iterative cycles of solution generation and solution evaluation (Bonnardel, 1989, Bonnardel, 1992), as designers continually refine their initial, abstract specification in order to reach a concrete solution (Hoc, 1987; Rasmussen, 1984).

However, the expression of evaluative judgments during design is difficult for several reasons:

- Designers often have incomplete evaluative knowledge.
- Designs need to be evaluated from multiple perspectives.
- Designers do not always recognise problematic solutions.

Incomplete Evaluative Knowledge. Not all designers are equally experienced; many newly hired or inexperienced designers lack detailed knowledge of the relevant criteria. Empirical studies have shown that experienced designers analyse conceptual or abstract designs using approximately twice as many criteria as inexperienced designers (Bonnardel, 1991). Some design domains are so large and complicated and have so many subdomains that no single person can know all there is to know. Thus, in complex domains, even experienced designers cannot know all the relevant criteria and constraints.

Evaluating from Multiple Perspectives. Multiple criteria and constraints can potentially be taken into account, but only some are relevant, depending on the perspective adopted. A perspective is a point-of-view which implies that certain design goals exist, certain bodies of design knowledge are relevant, and certain solution forms are preferred. Inexperienced designers may be unaware of the importance or even existence of other perspectives. Even experienced designers may forget to consider certain criteria useful for assessing features from a different perspective. Often times, design goals associated with different perspectives conflict and designers are forced to make difficult trade-off decisions. Even experienced designers often lack the background knowledge necessary for evaluating the trade-offs between different, competing solutions.

Recognising Problematic Solutions. Not all evaluation steps are explicit; many are implicit or tacit (Polanyi, 1966). Schön (1983) describes an action–breakdown–reflection cycle that underlies design practice. In this cycle, designers engage in situated action until their expectations are not met and they experience a breakdown in the current design solution. At that moment, designers stop and reflect on how to overcome the breakdown before proceeding. These breakdowns in situated action are opportunities for design evaluation. However, detecting and overcoming breakdowns requires much skill and domain-specific knowledge. People newly hired into a workplace will lack this necessary knowledge. Even experienced designers may not notice every possible problem.

Because of these difficulties, cooperative problem-solving systems (Fischer, 1990) that assist designers in constructing and evaluating design solutions could be beneficial. These systems are *cooperative* because the system helps users design solutions themselves as opposed to having an expert system design solutions for them. Such systems support design activities by allowing designers to creatively propose solutions, while the system actively helps designers to reflect on and assess various features of these solutions. The following section describes these systems in more detail.

3. Critiquing systems

Critiquing is a dialog where the interjection of a reasoned opinion about a product or action triggers further reflection on or changes to the artifact being designed. An agent – human or machine – capable of critiquing in this sense is a *critic*. Computer-based critics are made up of sets of rules or procedures for evaluating different aspects of a solution; sometimes each individual rule or procedure is referred to as a critic (Fischer et al., 1991, Fischer et al., 1993a, Fischer et al., 1993b).

Critiquing is integral to cooperative problem-solving systems. The human designer's primary role is to generate and modify solutions; the computer's role is to analyse these solutions and produce a critique for the designer to consider in the next iteration of this process. The core tasks of critics are to: (1) point out problematic situations that might otherwise remain unnoticed, (2) communicate debatable issues surrounding a design solution, and (3) help designers to learn about different views and perspectives. Critiquing systems *augment* the ability of human designers to evaluate their solutions; decisions concerning whether or not to follow the critic suggestions are always left up to the designers.

Designing a critiquing system that is both useful and cognitively ergonomic requires paying careful consideration to both the design of the system's interface and to the design of the system's knowledge-base. To a large extent, when evaluating the impact of a knowledge-based system, it is the content and structure of the knowledge-base that is being assessed. Thus, several aspects of design knowledge and practices need to be understood when creating the system:

- What are the evaluative knowledge rules (i.e., criteria and constraints) relevant to the specific domain being supported?
- What are the relevant evaluation procedures for assessing design solutions?
- How should the evaluative knowledge be partitioned or organised to reflect different perspectives in the domain?

The following subsection will review how these design issues were resolved in the Voice Dialog Design Environment. For a more detailed discussion of these and other issues concerning the design and construction of critiquing systems, see Fischer et al. (1993a, Fischer et al. (1993b) and Nakakoji et al. (1994).

3.1. The Voice Dialog Design Environment

Voice dialog designers have extensive training in Cognitive Psychology and Ergonomics or Human Factors (they usually have a Ph.D. in these areas). They are in charge of creating software applications with phone-based user interfaces. Typical applications are voice information systems and voice messaging systems. These applications consist of a series of voice-prompted menus requesting the user to perform certain actions; e.g., "to listen to your messages, press 1." The caller issues commands by pressing touch-tone buttons on the telephone keypad and the system responds with appropriate voice phrases and prompts. Designing in this domain means specifying the interface for an application at a detailed level. Depending on the size and complexity of the application being developed, this overall design process can take from six months to two years' time.

The Voice Dialog Design Environment (Sumner et al., 1991; Harstad, 1993) allows designers to quickly sketch out the flow of an audio interface by arranging domain-oriented design units such as voice menus and prompts into a flow chart-style representation (see Fig. 1). Designers can hear what their audio interface design sounds like by attaching audio recordings to components in the interface and simulating the design. Computational design critics embedded in the system watch designers' actions and comment on potentially problematic aspects of the design under construction.

3.2. Determining evaluative knowledge rules and procedures

Analysis of the designers' activities allowed us to identify that the designers were using two types of evaluative procedures: an *analytical* procedure for assessing the

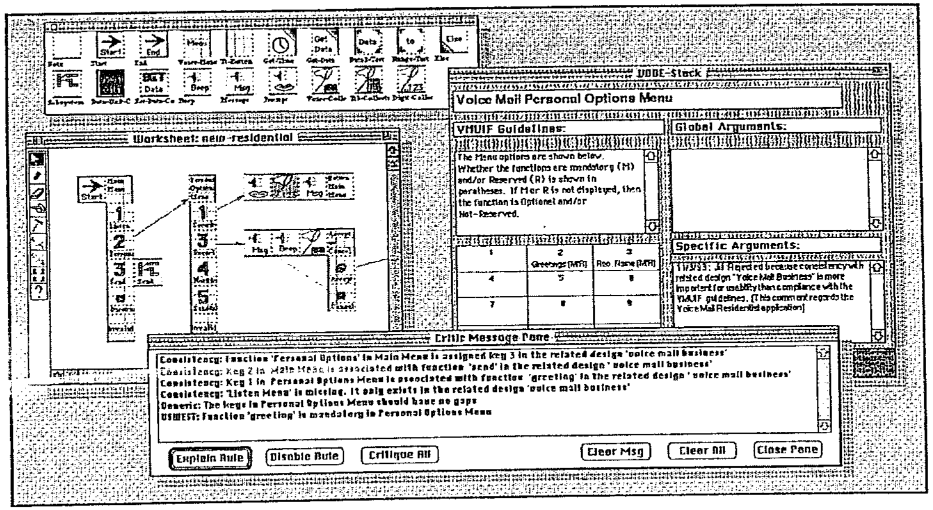

Fig. 1. The voice dialog design environment. Designers select building blocks from the gallery (top window) and arrange them in a worksheet (left window) to create a graphic representation of the audio interface design. A critiquing component analyses the design for compliance with interface guidelines and product consistency. Possible problems are signaled in the critic message pane (lower window). The designer can select a critic message and elect to see the design rationale behind the critic rule; designers can add to the design rationale in the hypermedia knowledge-base (right window).

solution's features with regard to criteria and constraints, and a *comparative* procedure for assessing the solution's features by comparison with the features of another solution (Bonnardel, 1992). We also identified that the designers could take into account several types of evaluative knowledge elements: criteria concerning *consistency* both inside and across design solutions as well as the criteria and constraints specified in various *user interface guidelines.* Analyses also showed that the relevance of some criteria depends on the *type of application* being developed. Therefore, we decided to embed in VDDE these two evaluative procedures and critic rules reflecting the various knowledge elements designers can potentially consider.

3.3. Partitioning the knowledge-base to reflect perspectives

One special challenge facing voice dialog designers is that their design task is influenced by many conflicting design objectives. From the perspective of satisfying new and novice end-users (i.e., users of phone-based products), these designers want to comply with different sets of user interface guidelines (regional, national, and international). However, to satisfy marketing groups and existing experienced end-users, designers also try to create designs that are consistent with related products or existing applications in the installed product base. The problem is that many applications in the installed base predate the interface guidelines and so do not conform to these guidelines. Thus, these design objectives are in conflict and the designer must make difficult trade-off decisions between these competing objectives.

The computational critics embedded in VDDE are intended to support designers to learn more about the different guidelines and the trade-offs between the different guidelines and product consistency. The knowledge-base is partitioned at the top level into four sets of critic rules that the designer can enable. Three sets of rules correspond to the regional, national, and international phone-based user interface standards. An analytical evaluation procedure is used to compare design solutions against these three rule sets. A fourth rule set – the consistency set – uses a comparative evaluation procedure to compare two designs for inconsistencies. Each rule set is further partitioned by application type; e.g., voice mail, call answering, voice message delivery, and voice bulletin board. Designers select and prioritise which rule sets to use during a design session. Designers can selectively enable more specialised evaluative knowledge by specifying the type of application being developed.

3.4. Designing with VDDE

Designers create designs by selecting units from the gallery and arranging them in the worksheets (Fig. 1). Designers can control the evaluations performed by the critiquing system in two ways. First, as described above, designers specify what evaluative knowledge (critic rule sets) should be enabled. Additionally, designers can also control the intervention rate of the critiquing system; i.e., the rate at which evaluations are performed.

If an active intervention strategy is selected, then the critiquing system automatically analyses the representation for compliance with all enabled rule sets whenever designers place or move design units in the worksheet. In addition, at any time the designer can order a complete assessment of the solution under construction causing all the solution's features to be checked, not just the ones resulting from the last actions made by the designer. This passive intervention strategy can be used either instead of, or in conjunction with, the active intervention strategy.

Regardless of the intervention strategy used, when a possible violation is detected, a brief message signaling the violation is immediately presented in a separate critic message window. To help designers to identify the source of a detected problem, each message is preceded by a symbol indicating the particular rule set the critic is part of.

The system provides deeper explanations of critiquing messages by linking the messages with portions of the relevant user interface guidelines documented in the hypermedia knowledge-base. At any time, the designer can add new information, such as rationale behind difficult trade-off decisions, to the hypermedia knowledge-base.

4. Study of the impact of VDDE on design activities

The design of VDDE is motivated both by a theory of design problem-solving and by an interpretation of the specific requirements of the domain of phone-based interface design. Our approach to assessing VDDE was targeted at these two different motivations. At the theoretical level, we wanted to assess the influence of system-provided evaluations on the design process. At the system level, we wanted to assess the quality of our particular domain knowledge-base design. Additionally, we wanted to compare how designers of different domain-skill levels used the system to characterise any differential effects the system might have.

4.1. Hypotheses

According to the *theoretical model of design problem-solving* presented in Section 2, problem-framing and problem-solving are influenced by successive solution evaluations. Thus, given the VDDE system's active evaluation interventions, we expected an effect on solution generation and problem-framing activities:

Hypothesis 1: Designers will modify design solutions and design specifications, in reaction to potential problems detected by the critiquing system.

Since less experienced designers may encounter more difficulties when evaluating design solutions than experienced designers, we also expected to see differential benefits based on the domain-skill level of the designers:

Hypothesis 2: The evaluation processes of less experienced designers will be more influenced by the critiquing system than the evaluation processes of experienced designers; specifically, less experienced designers will be presented with more critics and will take more actions in response to critic suggestions.

With regard to our objective to assess the *quality of our knowledge-base design*, we wanted to determine how the designers perceived the relevance of the VDDE system's evaluations. The perception designers have of the relevance of system-provided evaluations depends on both the content of critic messages and the timing of message presentation.

In accordance with Hypothesis 1, if the content and timing of the critic messages are relevant, they should lead designers to modify design solutions and/or design specifications. On the contrary, if designers judge the content of critics to be irrelevant to the current state of problem-solving they may either ignore critic messages, orally criticise these messages and/or record comments about them in the hypermedia knowledge-base.

In accordance with Hypothesis 2, less experienced designers may be more dependent on or influenced by system-provided evaluations (since they are less familiar with the specific guidelines and products used in the voice messaging domain), which leads to another potential effect based on the domain-skill level of the designers:

Hypothesis 3: Less experienced designers may argue less with the content of the critics presented by VDDE and, consequently, record in the hypermedia knowledge-base fewer comments or design rationale than the more experienced designers.

The main challenge for critiquing systems is to present critics that are both relevant to the task at hand and not intrusive. These two characteristics are related to the particular intervention strategy adopted. Passive critics, which are presented to designers only when they inquire, are obviously not intrusive but the main risk is that designers do not take into account important information when this information is useful (or the most useful). On the other hand, active critics, which are presented to designers as soon as a solution feature appears incompatible with certain critic rule(s), may be perceived as intrusive if they interrupt the designers' activities for reasons that are irrelevant or unimportant. In fact, preferences for different intervention strategies (passive or active) may depend on the domain-skill level of the designers:

Hypothesis 4: Less experienced designers may need active critics to guide them in their activities, whereas this strategy might be too intrusive for experienced designers.

4.2. Experimental situation

We constructed an experimental situation in which four professional designers were asked to think aloud while using VDDE to perform a single, realistic design task. In the paragraphs below, we describe the domain-skill level of the designers who participated in our experiment and the task they were presented with.

4.2.1. Participants

The four designers who took part in our experiment were experienced in the general principles of user interface design. All had done graduate work in either human–computer interaction or cognitive psychology; all had been professionally employed as phone-based interface designers by the same company. However, the important difference for this study was their level of experience in both general phone-based interface design and specifically, the design of voice messaging products:

- Two designers are specialised in the specific voice messaging products developed in the company, thus we consider them to be *highly-experienced* in the specific design area. We will refer to them as 'H-E designers'. They both have 11 years of experience in user interface design. One of them has been working in that specific design group for two years ('H-E designer 1'), and the other for about eight months ('H-E designer 2'). Both of these designers were familiarised with the VDDE system (they had previously used it to perform a few design tasks).
- The two other designers are less experienced in the specific area of voice messaging products. One has 3 years of experience in phone-based user interfaces, but not in the area of voice mail products, thus we consider him to have *medium-experience* ('M-E designer'). The other designer is specialised in user interface design but he worked only as a consultant for less than a year on phone-based products, thus we consider him to have '*low-experience*' in the specific area ('L-E designer'). Both of them were familiarised with the VDDE system through the solving of two design problems the day before performing the experimental task.

4.2.2. Experimental task

Our goal was to make the experimental task as real as possible, so we selected a design task representative of those usually performed in the voice messaging group. The requirements for this task were derived from the specifications of a product designed and marketed by another group in the same company, and were discussed and refined with the department head to ensure greater 'authenticity' (one participant commented at the end that the task was so real that she was somewhat overwhelmed initially).

The four designers who participated in this study were asked to integrate a new service into an existing voice messaging product using the VDDE system. They were given a one-page, textual requirements description that gave a little background on the existing product, the users in the installed base, and the main features to be integrated into the interface.

For this experiment, we chose to provide the designers with both active and passive critics. The active critics were automatically presented to the designers after an analysis of their last action; the passive critics were presented only when the designers asked for an assessment of their current design solution.

Participants were given 90 minutes to solve the design problem while thinking aloud. Due to the reality and complexity of the task, participants were told that we did not expect them to complete the design in the allotted time. During the course of the session, they could ask questions concerning the system, the critics, the task requirements, etc. Each design session was videotaped and the verbalisations of the designers transcribed. Approximately 70 pages of transcripts were analysed.

4.2.3. Post-experiment interviews

After the experimental task, a 20 to 30 minute open-ended, structured interview was conducted with each participant. The interview questions were targeted at getting designers to express how they perceived the influence of the critics on their activities and the relevance of the critics to the designers' task. We also asked the designers to explain how they perceived working with the system. To do so, they were provided with a list of four terms to choose from – spell-checker, collaborator, guiding hand, and watchdog – and were asked to explain why they selected a particular term.

4.2.4. Limitations of this approach

An evident limitation of our approach is that it is based on a small number of subjects. However, this frequently occurs when analysing the activities of professional designers in a specific workplace setting; there are simply not large numbers of these individuals to begin with. The benefit of using professionals in a naturalistic setting as subjects is that it allows us to obtain results that are directly related to the real-world situation being considered.

Another limitation is the lack of statistical analyses typically associated with classic experimental psychology methodology. Such analyses were not viable for several reasons. First, the small number of subjects involved in this study does not give us a meaningful sample size. Second, the open-ended and ill-structured nature of the realistic design problem afforded a large number of possible solutions of highly subjective quality; thus it would not be viable to simply statistically analyse design actions or design constructions in terms of quality or time to complete. Therefore, our study remains at an exploratory level; though the generalisability of the obtained results will be discussed in Section 5.

More importantly, our approach was based on our interest in studying 'cooperative' problem-solving between the designers and their design support system. To analyse the patterns of interactions, we needed to record the actions and reflections of *both* the designers and the VDDE system. Therefore, we decided to ask the designers to *think aloud* while performing the task. There are other potential limitations due to this method. Indeed, three types of drawbacks are usually noted (Caverni, 1988):

- verbalisation may modify the execution of the task (for example, see Ericsson and Simon, 1980, and Ahlum-Heath and Di Vesta, 1986);
- verbalisation does not allow access to all cognitive processes, especially the ones that have been automatised with experience; therefore, subjects may describe only partial processes or, even, processes that differ from what they really do (Visser, 1990);
- the interpretation of verbal protocols is subjective and this can be a source of errors (Hoc, 1992).

Though some of these limitations may apply to our situation, the specifics of our experimental situation limit the potential drawbacks with the protocol method. First, task modification may be limited if the task is already based on verbal activities. Some design theorists view design as a form of mental dialogue, noting that it is a "conversation with the materials at hand" (Schön, 1983) or an "argumentative process" (Rittel, 1984). Regardless of whether we can interpret these theories to indicate design is based on verbal activities (though mental), in our particular case, the designers all worked in design teams where group design sessions were common. Thus, they were all experienced in verbalising their design activities. Anyway, the potential side-effect of task modification will be discussed further in Section 5.

Access to key cognitive processes through verbalisations may be less limited in design activities since these problems are ill-defined and open-ended; the designer is always dealing with a problem that is, at least partially, new. Therefore, less processes become automatic and unaccessible through verbalisations.

Finally, a way to avoid errors due to the subjective interpretation and analysis of verbal protocol data is to have several analysts (or 'judges') separately perform the same analysis and then compare their findings. We used this approach and had four separate judges, each with graduate degrees in either cognitive psychology or human–computer interaction, analyse each of the transcripts.

4.3. Data analysis

First, a three-phase analysis was performed in order to *globally characterise the activities of the designers*:

1. After having read separately the verbal protocols several times, we agreed to divide the protocols into episodes corresponding to the main processes we identified in the design activity: understanding the task and the design problem, solution generation, and solution evaluation.
2. Next, events occurring in each episode were characterised according to:
   - *what* was discussed, generated or evaluated by the designers (e.g., solution elements or constraints);
   - *how* it was broached, generated (e.g., reuse of solution elements, analogical reasoning, etc.), or evaluated (analytical, comparative or global procedure – see Bonnardel, 1992);

- *which perspective or point of view* was adopted (the designer's, the future end-users, the marketing service, etc.).

3. Then, the order of occurrence of events and episodes was analysed.

Finally, to determine the *influence of the computational critics on the designers' activities*, we analysed in more detail the episodes in which designers were confronted with critics, particularly noting actions designers took in response to critic activity. Additionally, to characterise the relevance and intrusiveness of the critics, we supplemented the analysis with the comments made by designers during the post-experiment interviews.

4.4. Results

Analyses of the designers' activities and comments while using VDDE allow us to: (1) globally characterise the overall activities of the designers, (2) characterise the influence of computational critics on their reasoning and activities, and (3) make preliminary assessments concerning the relevance and intrusiveness of the critics.

4.4.1. Characteristics of the designers' activities

The succession of events and episodes during design may vary from one designer to another one. Indeed, this 'opportunistic' character of design often causes the design episodes for a single designer to continually oscillate between problem-framing and problem-solving (see Hayes-Roth and Hayes-Roth, 1979; Guindon, 1990b; Visser, 1990). However, at a general level, we identified two successive stages of design that were *common* to all the designers in this study:

1. A stage of understanding the task and construction of an initial mental representation of the problem, which can be viewed as an important part of the *'problem-framing'* performed by designers (see Section 2). Such a stage has been described by Akin (1978), using the term 'problem formulation' (or problem reformulation), as based on a process of information acquisition from the external environment and on processes of information interpretation and storage in memory of the interpreted information. Our analysis identified several sub-processes that are involved in this stage and contribute towards problem-framing:
   - analysis and assessment of prescribed constraints (i.e., constraints specified in the problem definition or specification);
   - analysis of the design's relationship to existing products referenced in the specification;
   - elaboration of scenarios envisioning what end-users will want to do with the final design product.
2. A stage of solution development or 'problem-solving' involving cycles of solution generation and solution evaluation. These two processes appear to be based on several sub-processes, which corroborate previous findings:
   - the taking into account of prescribed constraints;
   - the generation of new constraints and criteria (either 'deduced' from the current state of problem-solving or 'constructed' from the designers' own experience and preferences – see Bonnardel, 1989);
   - the reference to existing products, which underlies reuse (at least, partially) of previous design solutions and analogical or case-based reasoning (Kolodner, 1993);
   - the use of three different evaluative procedures (Bonnardel, 1992): a global procedure, when the designers assess globally a solution without referring to its specific features (e.g., "I think I have a good design", "this is really nice, I like that", "the print all baffles me"); an analytical procedure, when the designers assess a solution (or solution element) with regard to criteria and/or constraints e.g., "it is one of the worst in terms of complexity", "it took me [as the user] too long"); and a comparative procedure, when the designers assess a solution with regard to an alternative solution (e.g., "they both look really ugly");
   - internal, mental simulations of the functioning of the product, through verbal descriptions of the successive use of the product features or verbal simulation of interface components (see Guindon, 1990a);
   - external simulations of the functioning of the product using simulation features provided by VDDE (see Lebahar, 1983).

Thus, the design model presented in Section 2 seems to apply to the design of voice messaging products. However, though problem-framing also occurred during the stage of solution development, in this study, it seemed to occupy a more important role in the preliminary phases of design prior to using VDDE. Designers constructed an initial representation of the task and problem, before trying to solve it with the support of VDDE, that didn't undergo much apparent change during the course of design activity. In addition, this global analysis enables us to point out a first result linked to the domain-skill level of the designers: during this first stage, more-experienced designers (i.e., the two H-E designers) *assessed more prescribed constraints* than the less-experienced designers (i.e., M-E and L-E designers). The highly experienced designers did not hesitate to point out the deficiencies of various problem requirements and, even, decided to modify requirements they did not like.

4.4.2. Impact of the critiquing system on the designers' activities

The impact of the critiquing system on the designers' activities was analysed with regard to the hypotheses previously presented. First, we expected a global effect from using VDDE, i.e., that critics would lead designers to modify their solutions and/or problem specifications (Hypothesis 1). We were also expecting that the benefit of the system would be more important for less-experienced designers and, specifically, that they would be presented with more critics and would take more actions in response to critic messages (Hypothesis 2). Therefore, we analysed both quantitatively and qualitatively the critics considered by the designers, and we characterised the consequences of the critics on the designers' actions and reasoning.

(I) Number of critics taken into account by designers

A first result is that *the L-E (low-experienced) designer was presented with more critics than the M-E and H-E designers* (respectively 15 vs. 6 or 3 critics – see Table 1). In addition, we observed that the H-E designer who had the longest experience in voice

Table 1
Number of presented and anticipated critics according to the designers' domain-skill level

	Presented critics	Anticipated critics	Total no. of critics
L-E designer	15	1	16
M-E designer	6	3	9
H-E designer 2	6	3	9
H-E designer 1	3	2	5

messaging products was presented with the least number of critics. Such results corroborate Hypothesis 2.

A second result was unexpected: the critiquing system seemed to exert an indirect effect on the designers' reasoning. Thus, it appears that *all designers* (even the low-experienced designer) *anticipated critics*. Such anticipations were expressed through comments such as "I wonder if the system will catch me" or "I know I'm going to get dinged here." Some anticipations were confirmed by the later presentation of critics, others were not (this is discussed more fully below). Therefore, the influence of the use of a critiquing system seems to be also 'indirect': *just the fact of knowing that the critics could fire, influences the designers' reasoning to look for deficiencies in their solutions that might cause the presentation of critics by the system.* Such an anticipation of critics implies that the designers knew *before the presentation of the critics* that they were going to break design rules that may be embedded in the critiquing system.

The interpretation of the ability of the designers to anticipate critics may be different for the H-E designers and the L-E and M-E designers. The two H-E designers probably had previous knowledge concerning what design rules might be represented in the critiquing system due to their greater design experience. In the case of the L-E and M-E designers, this finding can be interpreted differently: it may suggest that (1) these designers also had previous knowledge of some general phone-based interface or voice messaging-specific design rules, or (2) they had learned these design rules while using the critiquing system (either during this experimental session, or during the two preparatory sessions). To investigate further these two interpretations, we analysed the nature of both the presented and anticipated critics.

(II) Nature of the critics

The analysis of the *presented critics* allowed us to distinguish three types of presented critics according to their content:

- general design rules, i.e., independent of the specific voice mail application (such as "keys should have no gaps" or "should not use more than 4 items");
- specific design rules, concerning the terminology or command names occurring in a voice mail application or about the assignment of functions to different keys ("cancel should be assigned the * key");
- internal consistency of the design solution (i.e., among the different menus of the current solution) or external consistency (i.e., between the current design solution and a preexisting solution).

The analysis of the *anticipated critics* showed that only H-E designers were able to anticipate critics they had not been presented with during the current design session. Both the M-E and L-E designers appeared to anticipate only critics that had been previously presented during the current session; which indicates that these designers learned 'in action' critic rules, while using VDDE to solve the proposed design problem.

(III) Consequences of the presented and anticipated critics on the designers' activities

Whether the critics were actually presented or only anticipated, we observed a common reaction from *all designers*: designers *assess the relevance of critics* before developing an action to improve the design solution. The assessment is immediate for critics that are related to specific design rules or to consistency issues, and appeared to usually lead to solution modifications.

Contrarily, critics reflecting general design rules lead designers either to look for further information before deciding whether to apply them or to immediately reject them. Regardless of designers' experience level, the primary cause of rejection was in conflict with the other design goal of product consistency. In many cases, creating consistent interfaces requires breaking many general design guidelines and this source of conflict is endemic to the nature of design in this particular domain.

Therefore, the assessment of either the presented or anticipated critics can lead designers to *two alternative decisions*:

1. The decision to follow a critic's advice immediately or after a delay (in order to look for further information) and either modify a solution feature or, more rarely, a problem specification (only one time from a H-E designer). We observed that solution modifications were limited to mainly surface features such as renaming an option in a menu or changing the key assigned to an option.
2. The decision not to follow the critic's advice for various reasons:
   - Many times designers considered the content of the critic to be correct, but not to apply to their specific design case. This cropped up frequently in conjunction with conflicts between consistency and usability design objectives. (ex: the designer reads the presented critic: "keys in fax options should have no gaps", and reacts to it: "but they have to for consistency");
   - In a few cases, designers argued with the correctness of the rationale behind the considered critic (ex: the designer reads the presented critic: "first item should be 1", and reacts to it: "yes, except ... this looks like a problem within the [company name] rule itself, because these are like special cases of menus, where 1 does not necessarily have to be", which leads him to add his own comments in the hypermedia knowledge-base of the system);
   - Sometimes designers were presented with critics they had already decided not to follow and had not disabled, thus these critics were presented again (ex: "I have already taken care of that").

With regard to Hypothesis 1, our results do not show a noticeable impact of critics on problem specification activities since designers rarely reformulated the design problem in reaction to critic messages. The results do corroborate, in some cases, the hypothesis that critics lead to design solution modifications.

In addition, we observed that the consequences of the presented and anticipated critics depended – to a certain extent – on the *domain-skill level of the designers*:

1. In reaction to *presented critics*, less-experienced designers seemed to modify slightly more solution features than more-experienced designers (3 solution modifications by L-E and M-E vs. 1 by each of the two H-E designers). Thus, our results tend to corroborate Hypothesis 2: less-experienced designers will take more actions in response to critic suggestions.
2. In reaction to *anticipated critics*, both the more- and less-experienced designers modified solution features, but the H-E designers appeared to *analyse more deeply* the reasons why they were planning to break certain design rules. In a few cases, this leads to modifications of the problem requirements:
   Example excerpt from a designer's verbalisation:
   - *Anticipation of a critic*
     "I'm struggling with a critic I know is going to come up. I've got four options here... and that's the maximum number I should have on the menu. But I also need to give them [the end-users] a way to listen to, so I'll need a key for that... So, I've got five options, and that's violating the design guideline."
   - *Analysis of the reasons for breaking the critic rule*
     "I think the reason I backed myself into this corner is I have picked up on the way our current voice messaging vendor has integrated this [service] into their product [...] I knew I was going to get an error because I was violating design guidelines since I have too many options in the menu ..."
   - *Attempt to modify the design in order to satisfy the critic rule ... which leads to modifying a design requirement*
     "I don't like the way the menu looks. It's a good guideline. So at this point, [...] I think I would argue for removing the option to print the old documents. [...] I would renegotiate with the market unit about the need to get one of them taken out. And then I'd be able to have everything fit on the menu."

4.4.3. Relevance and intrusiveness of critics

A challenge in building critiquing systems is to provide designers with *critics that are relevant to their current tasks, without being intrusive*. These two criteria are in fact interdependent: critics are usually considered non-intrusive when they are perceived to be relevant to the task at hand. The relevance of the critics depends both on their content and on the moment at which they are presented; a critic can be informative at a certain step during problem-solving but uninformative before and after this step.

Concerning the *content* of critics, we observed that contrarily to Hypothesis 3, both the more- and less-experienced designers orally disagreed with certain critics. When they decided not to follow critics, these designers sometimes recorded comments into VDDE's hypermedia knowledge-base. These records allow the designers to express that they perceived critic contents to be incorrect (this occurred only once from one designer) or, more frequently, to be correct but to not apply to the specific design case.

Another factor that could influence the perception of relevance is the critics' *level of abstraction*. The critics provided by VDDE (as well as by other critiquing systems) are based on an analysis of specific components of the solution, situated at a low level of abstraction (e.g., they can assist the designers to deal with issues such as "should button 6 be 'replay'?"), whereas the designers' concerns are often at a higher level (e.g., should 'replay' be in this menu? what should the features of the product be?). Thus, one of our assumptions was that critics at a radically different abstraction level from the designer's current thinking could be perceived as jarring or irrelevant. Such a problem occurred more with the less experienced designers than for the highly experienced designers. For highly experienced designers, even critics at a low level of abstraction appeared to motivate them to develop a deep analysis of their state of problem-solving. In addition, when asked to comment on the presented critics, H-E designers appeared to have basically not noticed or cared about this difference in abstraction level. One possible explanation is that these designers have deep domain knowledge, which allows them to effortlessly bridge this abstraction gap. For example, when one of the designers modified the design specification, the surface level issue was too many menu items, but she immediately mapped it to a deeper structural issue of unnecessary printing options in the design specification.

As pointed out before, the perception of appropriate (or not) *timing* may be dependent on the intervention strategies of the critics:

(1) The risk associated with *active critics* is that they fire continually and, therefore, become uninformative and intrusive for the designers. In our study, we observed that relatively small numbers of critics were presented to both the less- and the more-experienced designers. This finding differs significantly from other studies of critiquing systems (Nakakoji, 1993). One explanation could be the domain-skill level of the designers. In previous studies, primarily novice subjects with no domain-skill experiences were used. In this study, these designers had some previous knowledge of design guidelines; they applied this previous knowledge towards creating a good design and thus did not break many rules. As previously discussed, designers also anticipated critic activity; thus when an anticipated critic finally fired, it was expected and *not considered intrusive*. Such a finding seems to go partially against Hypothesis 4: though less experienced designers seem to need active critics to guide them in their activities (as pointed out with Hypothesis 2), the active strategy does not appear to be too intrusive for experienced designers.

(2) The risk associated with *passive critics* is that they are activated too late in the design activity, when designers are too committed to an inadequate design orientation. We did not observe problems of this sort in this study. However, we did observe differences in the use of passive critics: H-E designers often forgot to activate passive critics at all, whereas less-experienced designers activated passive critics periodically when they felt they had completed a design subtask. It appeared that the less-experienced designers, who might not feel very secure about their design, used passive critics to verify that they had not made any glaring errors or missed important features. While conclusions are difficult to draw based on a single study, in post-session interviews the less-experienced designers characterised the influence of the critiquing systems as being more similar to a design 'spelling-checker'. This characterisation seems to be in accord with their observed use of passive critics.

Table 2
Summary of results

	Hypotheses	Results	Unexpected findings
Impact of critics on designers' activities			
Hypothesis 1 (global effects)	• Part 1: Critics will lead to changes in solution features • Part 2: Critics will lead to changes in problem specifications	• Observed but → • Rarely observed	• *Assessment of critics by all the designers* Decisions dependent on the nature of the critics (general vs. specific to voice messaging products)
Hypothesis 2 (partial effects)	• Less-experienced designers will be presented with more critics and will take more actions in response to critic messages	• Observed	• *Anticipation of critics* by all the designers • *More comprehensive reflection* by the more experienced designers
Relevance and intrusiveness of critics			
Hypothesis 3 (partial effects)	• Less-experienced designers will argue less with the content of critics and add less comments into the hypermedia-knowledge base	• Unobserved	• *Further analysis of the critics' content* by the more experienced designers
Hypothesis 4 (partial effects)	• Less-experienced designers need active critics, contrary to more-experienced designers	• Unobserved	• Active critics seem *fine for all designers*

5. Discussion

In this section, we first comment on the results we obtained during this study and contrast them with results also obtained at the University of Colorado but with non-professional designers interacting with another critiquing system. Then, we reflect on the benefits of using critiquing systems according to the domain-skill level of designers, and we finally consider again the main limitation of our study and discuss ways to address it.

5.1. Comments on the obtained results

The results we obtained tend to corroborate certain hypotheses and, more interestingly, they showed some unexpected findings about the designers' activities (see Table 2).

In accordance with part 1 of Hypothesis 1, designers modified solution features ir reaction to critics; but contrary to part 2, they rarely modified design specifications Such a result can be compared with the results of another study (Nakakoji, 1993) conducted also at the University of Colorado. This study used a critiquing systerr targeted at architectural design, particularly the design of kitchen floor plans anc subjects who were not professional designers (students at the University). Subjects were asked to use a critiquing system (the KID system, see Nakakoji, 1993) to design a kitchen. Similar to our study, Nakakoji reports an 'asymmetry in the coevolution o design specification and design construction'. Nakakoji observed that subjects modifiec design constructions much more than design specifications in response to critiquin messages.

An explanation based on the global analysis of design activities can be offered fo these results. This analysis showed that designers were focused on the construction of a task or problem representation at the beginning of their activities, i.e., before usin VDDE. Indeed, we observed that it was during this first stage that designers (especiall the more-experienced ones) assessed and modified requirements. Therefore, thoug designers refine their problem representations during successive problem-solving steps the changes made seem limited to the definition of new constraints and criteria.

In accordance with Hypothesis 2, less-experienced designers were presented wit more critics than more-experienced designers, and followed slightly more the critics suggestions. However, an unexpected finding was that *all designers assessed the critic* they were presented with before deciding whether to follow the critics' advice. Thei decisions appeared to be dependent on the nature of the critics: they usually followe critics representing specific design rules but not the ones representing general desig rules. This probably reflected the fact that they gave consistency with related products a slightly higher priority than conformance to user interface design guidelines. In addition we observed differences in the reactions of more- and less-experienced designers t critics representing specific design rules: though these critics are situated at a low leve of abstraction, *the more-experienced designers interpret the criticism at a higher leve and develop a more comprehensive reflection* than the less-experienced designers do.

Contrary to Hypothesis 3, less-experienced designers do not seem to argue less wit critics than more-experienced designers, nor to add significantly less comments in th hypermedia knowledge-base. They all react orally to critics they disagree with and sometimes, in writing by adding comments into the hypermedia knowledge-base However, in accordance with the previous observation, the more-experienced designer analyse further the content of the critics they react to and tend to point out exceptions t the critic rules.

Finally, contrary to Hypothesis 4, an active critic intervention strategy seems to b fine for all designers. Indeed, all designers were presented with a relatively low numbe of critics. This result is explained by the fact that *all the designers anticipated critic and, therefore, were able to avoid some of them when they wished.* This seems to show that the more-experienced designers rely on knowledge they have gained in othe situations, and that the less-experienced designers learned design rules while using th VDDE system and were able to apply them to later stages of problem-solving.

Table 3
Possible impacts and benefits of the critiquing system on the activities of designers, according to their domain-skill level

Domain-skill level	Inexperienced designers	Low-experienced designers	High-experienced designers
Impacts of the use of a critiquing system	· provided with numerous critics · frequently follow the guidance of the critics · try to reach an 'ideal' solution	· provided with less critics · less frequently follow the guidance of the critics · anticipate some critics · assess critics before applying them and do not try to satisfy all of the critcs	· rarely provided with critics · rarely follow the guidance of the critics · anticipate more critics · assess critics before applying them and do not try to satisfy all of critics · interpret critics at a higher level of abstraction and reflect more on the critical issues
Overall benefits observed	· reach a better solution (possibly other benefits)	· reach a better solution · learn critic rules	· reach a better solution · get an increased awareness of difficult trade-off decisions

5.2. Benefits of critiquing systems according to the designers' level of experience

Who are critiquing systems for, novice designers or experienced professionals? The combination of Nakakoji's (1993) results and ours provide some answers to this question. Critiquing systems may benefit both inexperienced designers and experienced designers (wherever they are situated on a spectrum of experience), but in different ways (see Table 3):

The benefit for inexperienced designers is *direct*: they follow the guidance provided by the critics and modify mostly solution features and, occasionally, problem requirements. The benefit for experienced designers is more *indirect*: the fact of using a critiquing system leads them to anticipate critics and, by doing so, prompts them to analyse in depth why they choose to break design rules. However, such a benefit appears to be more important for more-experienced designers, since they interpret the critics at a more abstract level than the less-experienced designers do. This finding was confirmed by the designers' comments during the post-experiment interviews: the more-experienced designers rated the impact of the critics on their activities higher than less-experienced designers.

Another difference lies in the fact that both the more- and less-experienced designers in our study appeared to *assess* both anticipated and presented critics in order to decide whether they will apply them. This finding differs from Nakakoji's study using inexperienced designers; these novices tried to reach an 'ideal' solution which satisfied

all critic rules and, therefore, perceived their task as almost impossible since many of th design rules conflicted. In addition, the more experienced subjects in this study ha previous knowledge of some of the design rules, thus their activities were les interrupted by critics than the designers in Nakakoji's study. These results tend t suggest that active critics can be used with experienced designers – wherever they ar situated on a spectrum of expertise – whereas a filter would be desirable for inexperi enced designers in order to point out, at least initially, only major drawbacks in thei design solutions.

Overall, these findings have implications for our knowledge-base design goal o increasing designers' awareness of difficult trade-off decisions between competin design objectives. In this area, it appears that the critiquing system has differentia benefits; designers with greater domain-skill level get most of the benefit. Makin difficult trade-off decisions involves questioning design rules, breaking the rules anc sometimes, modifying initial problem specifications. In this study, only highly experi enced designers engaged in these activities in response to critic influences. In Nakakoji' study, inexperienced designers rarely modified problem specifications and felt uncom fortable breaking design rules. These results seem to indicate that the influence of criti activity on designers with less domain-skill experience is not enough to get them t question or modify initial problem specifications and design goals. Thus, the critiquin system appears to have limited influence on less experienced designers when it comes t increasing their awareness of difficult trade-off decisions.

5.3. Limitations and future work

One of the limitations discussed previously still remains and may cause a side-effec the thinking aloud instruction may make designers reflect more on what they are doin (Ericsson and Simon, 1980). If such a side-effect was occurring, it would happen fo both the more- and less-experienced designers who participated in our study and it coul explain, at least in part, their common ability to anticipate and assess critics. Howeve the differential results of our study indicate that the observed results cannot be entirel due to the thinking aloud instruction. Also, the ability of designers to anticipate an assess critics was not observed in Nakakoji's (1993) study, though she also aske participants to verbalise their thoughts.

A supplementary experimental situation could be built to determine whether th ability to anticipate and assess critics is due to the use of the critiquing system or to th method of verbalisation. For instance, an experimental situation could be constructe with a group of subjects having respectively the same domain-skill level as the designer in this study. These designers would use VDDE without the critiquing system and onl be given the guidelines and other written documents that had been used to construct th VDDE knowledge-base. In this case, the side-effect due to the thinking aloud instructio would still apply (and prompt these designers to anticipate and assess design rules; i.e. 'critics'). However, if VDDE is really influencing the anticipation and assessment o critics, designers who used the system to perform the task should anticipate and asses more critics than the designers not using the critiquing system.

Such a situation would also allow us to determine the impact of VDDE on th learning of critic rules by designers. Indeed, the comparative situation described above could demonstrate that one of the major benefits of this system may result from the fact that it provides designers with contextualised information, directly related to the designers' task at hand. Therefore, the designers can learn knowledge in a functionally organised manner (i.e., with regard to the types of problems it allows the designers to solve), contrary to the rationally organised knowledge usually presented in classes or professional literature (see Bastien, in press).

Finally, the only way to confirm the impact of VDDE on design activities would be to conduct a long-term comparison of designers both using and not using the system. Such a comparison requires designers to be selected that have comparable educational and professional backgrounds. This would enable us to progressively characterize the increase in their domain-skill level and, thus, determine whether the use of VDDE helps designers to develop, either more quickly or in more depth, abilities that are naturally developed when humans acquire experience in a specific domain.

However, when we choose to study real-world situations in Cognitive Ergonomics, the realisation of studies such as the ones described above remains particularly difficult (or, even impossible). When performing studies in naturalistic settings, we can only manipulate a very limited number of factors. As researchers, we have to make difficult trade-offs between performing studies that conform to our theoretical objectives versus those that provide results relevant to the rich and varied real-world context.

6. Summary

The work we have presented in this paper is three-fold, consisting of a theoretical description of design activities, a practical description of a computational system that supports evaluation in design ('critiquing systems'), and a study of the influence of this system on designers of different domain-skill levels. The contribution of this research is both theoretical and practical. On a theoretical level, it contributes towards a better understanding of the evaluation processes used by professional designers. At a practical level, it contributes towards our understanding of how to create more useful and usable knowledge-based computer systems.

Overall, this study found both some expected and unexpected results. Some of the unexpected results indicate possible areas of the critiquing system that are problematic and suggests improvements to be incorporated into the next generation of design environments. However, other unexpected results, such as the finding that designers anticipate critic messages, could be endemic to cooperative problem-solving systems in general. Thus, these findings contribute on a broader level to our understanding of human–computer interaction, and particularly the possible impact of complex systems on the cognitive functioning of the systems' users. It is these unexpected results that highlight the increasing need for cognitive ergonomic studies such as this as cooperative problem-solving systems become more prevalent in the workplace.

Acknowledgements

We thank the designers at U S WEST Advanced Technologies and members of the Center for LifeLong Learning and Design for their help and collaboration during this project. We also wish to thank Floris van Nes, John Rieman, and two anonymous reviewers (for commenting on drafts of this paper), Benedikte Harstad (who developed the critiquing component of VDDE), Anne Biancato and Rémi Mouttet (who participated in the data analyses), Michael Wright (who helped with the preparation of this manuscript), and Marie-Julie (who waited so patiently). This research was supported by U S WEST Advanced Technologies, ARPA (under grant no. N66001-94-C-6038), and the Programme Cognisciences of C.N.R.S.

References

Akin, O., 1978. 'How do architects design?' In: Latombe (ed.), Artificial intelligence and pattern recognition in computer aided design. Amsterdam: North-Holland.

Ahlum-Heath, M.E. and F.J. Di Vesta, 1986. The effect of conscious controlled verbalisation of a cognitive strategy on transfer in problem solving. Memory and Cognition 14(3), 281–285.

Bastien, C., in press. 'Raisonnement, résolution de problèmes et contrôle de l'activité'. In: R. Ghiglione and J.F. Richard (eds.), Cours de psychologie, Tome III. Paris: Dunod.

Bonnardel, N., 1989. L'évaluation de solutions dans la résolution de problèmes de conception [The evaluation of solutions in design problem-solving]. Research Report No. 1072, INRIA, Rocquencourt.

Bonnardel, N., 1991. 'Criteria used for evaluation of design solutions'. In: Y. Queinnec and F. Daniellou (eds.), Designing for everyone and everybody. Proc. 11th Congress of the International Ergonomics Association (pp. 1043–1045). London: Taylor and Francis.

Bonnardel, N., 1992. Le rôle de l'évaluation dans les activités de conception [The role of evaluation in design activities]. Ph.D. dissertation, University of Provence.

Burkhart, B., D. Hemphill and S. Jones, 1994. The value of a baseline in determining design success. Human Factors in Computing Systems (CHI '94), 386–391.

Caverni, J.-P., 1988. 'La verbalisation comme source d'observables pour l'étude du fonctionnement cognitif'. In: J.-P. Caverni, C. Bastien, P. Mendelsohn and G. Tiberghien (eds.), Psychologie cognitive: Modèles et méthodes (pp. 253–273). Grenoble: Presses Universitaires de Grenoble.

Eastman, C.M., 1969. Cognitive processes and ill-defined processes: A case study from design. First Joint International Conference on A.I. (pp. 669–690).

Eisenberg, M. and G. Fischer, 1994. Programmable design environments: Integrating end-user programming with domain-oriented assistance. Human Factors in Computing Systems (CHI '94), 431–437.

Ericsson, K.A. and H.A. Simon, 1980. Verbal reports as data. Psychological Review 87(3), 215–251.

Fischer, G., 1990. Communication requirements for cooperative problem solving systems. Informations Systems 15(1), 21–36.

Fischer, G., 1994. 'Domain-oriented design environments'. In: L. Johnson and A. Finkelstein (eds.), Automated software engineering (pp. 177–203).

Fischer, G., A.C. Lemke, T. Mastaglio and A.I. Morch, 1991. Critics: An emerging approach to knowledge-based human computer interaction. International Journal of Man–Machine Studies 35(5), 695–721.

Fischer, G., K. Nakakoji, J. Ostwald, G. Stahl and T. Sumner, 1993a. 'Embedding computer-based critics in the contexts of design'. In: S. Ashlund, K. Mullet, A. Henderson, E. Hollnagel and T. White (eds.), Human Factors in Computing (Interact '93 and CHI '93), 157–164.

Fischer, G., K. Nakakoji, J. Ostwald, G. Stahl and T Sumner, 1993b. Embedding critics in design environments. The Knowledge Engineering Review 8(4), 285–307.

Gould, J.D., 1980. 'Experiments on composing letters: Some facts, some myhths, and some observations'. In: L.W. Gregg and E.R. Steinberg (eds.), Cognitive processes in writing. Hillsdale, NJ: LEA.
Guindon, R., 1990a. Knowledge exploited by experts during software system design. International Journal of Man–Machine Studies, Special Issue: What Programmers Know 33(3), 279–304.
Guindon, R., 1990b. Designing the design process: Exploiting opportunistic thoughts. Human–Computer Interaction 5, 305–344.
Guindon, R., 1992. Requirements and design of design vision, An object-oriented graphical interface to an intelligent software design assistant. Human Factors in Computing Systems (CHI '92), 499–506.
Harstad, B., 1993. New approaches to critiquing: Pluralistic critiquing, consistency critiquing, and multiple intervention strategies. Masters Dissertation, University of Colorado at Boulder, Computer Science.
Hayes-Roth, B. and F. Hayes-Roth, 1979. A cognitive model of planning. Cognitive Science 3, 275–310.
Hoc, J.-M., 1987. Psychologie cognitive de la planification. Grenoble: Pressses Universitaires de Grenoble.
Hoc, J.-M., 1992. L'extraction des connaissances et l'aide à activité humaine. Intellectica 12, 33–64.
Kolodner, J., 1993. Case-based reasoning. San Mateo, CA: Morgan Kaufmann.
Lebahar, J.-C., 1983. Le dessin d'architecte. Roquevaire: Parenthèses.
Nakakoji, K., 1993. Increasing shared knowledge between humans and design environments: The role of a specification component. Ph.D. dissertation, University of Colorado at Boulder, Computer Science.
Nakakoji, K., T. Sumner and B. Harstad, 1994. 'Perspective-based critiquing: Helping designers cope with conflicts among design intentions'. In: J. S. Gero and F. Sudweeks (eds.), Artificial Intelligence in Design '94, 449–466.
Pavard, B., 1985. La conception de systèmes de traitement de textes. Intellectica 1(1), 37–63.
Polanyi, M., 1966. The tacit dimension. Garden City, NY: Doubleday.
Quinn, J. B., 1992. Intelligent enterprise. New York: The Free Press.
Rasmussen, J., 1984. Strategies for state identification and diagnosis in supervisory control tasks, and design of computer-based support systems. Advances in Man–Machine Systems Research 1, 139–193.
Reitman, W.R., 1965. Cognition and thought: An information processing approach. New York: Wiley.
Rittel, H.W.J., 1984. 'Second-generation design methods'. In N. Cross (ed.), Developments in design methodology (pp. 317–327).
Schön, D.A., 1983. The reflective practitioner: How professionals think in action. New York: Basic Books.
Simon, H.A., 1981. The sciences of the artificial (2nd ed.). Cambridge, MA: The MIT Press.
Sumner, T., S. Davies, A.C. Lemke and P.G. Polson, 1991. Iterative design of a voice dialog design environment. Technical Report No. CU-CS-546-91, University of Colorado at Boulder, Computer Science.
Visser, W., 1990 More or less following a plan during design: Opportunistic deviations in specification. International Journal of Man–Machine Studies, Special Issue: What Programmers Know 33 (3), 247–278.
Whitefield, A., 1986. 'An analysis and comparison of knowledge use in designing with and without CAD'. In: A. Smith (ed.), Proceedings of CAD. London: Butterworth.

INSTRUCTIONS TO AUTHORS

Manuscripts should be submitted in quadruplicate to: *M.W. van der Molen, Faculty of Psychology, University of Amsterdam, Roetersstraat 15, 1018 WB Amsterdam, The Netherlands.*

Letters to the Editor should not exceed two printed pages. Short reports not exceeding three printed pages can be published as well.
Papers must be in the English language and conform to the highest standards of presentation.
Electronic manuscripts. Submission of accepted papers as electronic manuscripts, i.e., on disk with accompanying manuscript, is encouraged. Please make absolutely sure that the file on the disk and the printout are identical.
Manuscripts should be printed on one side of the page only, double-spaced and with a wide margin. Pages should be numbered consecutively. The cover page should contain: (i) the title of the article, (ii) author(s), (iii) complete affiliation(s), and (iv) e-mail address, fax and telephone number of the corresponding author.
An **Abstract** should be provided conforming to the "Outline for Preparation of Abstracts" in Psychological Abstracts (PA), to be printed at the beginning of the paper. Following the Abstract one to five **Keywords** (from the American Psychological Association's (APA) "Thesaurus of Psychological Index Terms") and at least one **PsycINFO Classification code** (from the APA's "PsycINFO Classification Categories and Codes") should be added. Assign code(s) by mentioning "PsycINFO classification", to be followed by the four digits – code(s) preceding the relevant classification category(ies).
Titles and subtitles should be numbered. If the paper is an experimental one, it should be divided, for each experiment, into appropriate headings like: Method, Results, Discussion/ Conclusions.
References. In the text of the manuscript, reference to a publication should be made by the name of its author, followed by the year of its publication between parentheses, thus: Miller (1990) found that ..., or ... as studied previously (Miller, 1990). The complete references must be given on a separate list, arranged alphabetically with respect to the first author's name. In the case of a publication brought out jointly by three or more authors, it should be quoted in the text as Miller et al. (1990); in the list, however, all names must be given in full. If an author or a particular group of authors have brought out several publications in one year, the corresponding references should be distinguished in the text as (1990a, 1990b) and in the list as 1990a, 1990b. *Examples* (please note that journal titles should not be abbreviated):
for journal articles: Miller, J., 1990. Discreteness and continuity in models of human information processing. Acta Psychologica 74, 297–318.
for books: Stelmach, G.E. and J.A.S. Requin (eds.), 1980. Tutorials in motor behavior. Amsterdam: North-Holland.
for contributed volumes: Sergent, J., 1989. 'Microgenesis of face perception'. In: A.W. Young and H. Ellis (eds.), Handbook of research on face processing. Amsterdam: Elsevier.
Figures. Figures should be large-size originals (each on a separate sheet), drawn in India ink and carefully lettered, or should be produced using professional quality graphics software and a laser- or equivalent printer. They should have an arabic number and a caption. In the text, figures must be referred to as: see Fig. 1; or Figs. 2 and 3, etc. Their approximate location in the text should be indicated as follows:

Insert Fig. 1 about here

Tables. Tables must be typed on separate sheets and should have a short title and an arabic number. The reference to tables in the main text and the indication of their approximate location is the same as for figures.
Formulae in the text and mathematical symbols. Avoid superposition of symbols (fractions or complicated exponents) which would necessitate a greater space between the lines; when superposition cannot be avoided place the expression on a separate line. Decimal numbers should have a zero before the decimal point, thus: 0.05.