Gouvernement du Canada
Ministère des Communications

Government of Canada
Department of Communications

Le Centre canadien de recherche sur l'informatisation du travail
Canadian Workplace Automation Research Centre

User-friendliness of interfaces: Proposal for an Evaluation Methodology

by

François Lapointe

Canada

Communications Canada
Canadian Workplace Automation Research Centre
Organizational Research Directorate

User-friendliness of interfaces: Proposal for an Evaluation Methodology

by

François Lapointe

Working Paper

Laval
August 1990

Également disponible en français sous le titre
La convivialité des interfaces :
proposition d'une méthodologie d'évaluation

This report is conducted by Organizational Research Directorate of the Canadian Workplace Automation Research Centre of Communications Canada. The opinion it contains are those of the author only.

Cat. No. Co28-1/75-1991E
ISNB 0-662-18585-4
(Original edition ISBN 0-662-96677-5)

ACKNOWLEDGMENTS

The author wishes to thank Judith Légaré and Richard Lavoie, researchers in the evaluation field with the Organizational Research Directorate of the Canadian Workplace Automation Research Centre. Their comments and suggestions during the drafting of the paper were of great assistance.

TABLE OF CONTENTS

INTRODUCTION

The past decade has seen a substantial broadening both of computer applications and of the population of computer tool users. The result is a diversification of needs, levels and types of technical skills, as well as of the "professional cultures" of user individuals and organizations. As a consequence of their success, computer tools must now be able to dialogue with numerous, diverse partners. This diversity of applications and users makes the aims and challenges of the computer design and installation process more complex. The choices to be made now no longer turn merely on the purpose of a tool operated by experts. They also concern the way in which this tool is used and the nature of the expertise, both professional and technical, of the users who will have to work with it. In this context, the question of the tool-user interface[1] becomes a prominent one.

A field of inquiry still in its infancy, the study of the social processes accompanying the introduction of computer tools in the workplace has barely begun to reveal the complexity of the relationship among the tool, the task to be performed, the user and the organizational dynamics encompassing them. The concept of user-friendliness of the computer tool-user interface lies within this complex relationship.

---

1. When we talk of interface, it will be in a broad sense, referring not only to user-tool communication channels (as used by Graham Storrs, 1989), but also to the **set of interaction components** (psychological, organizational and circumstantial) associated with useing these channels.

Indeed, tool-user interaction may be defined as user-friendly only in relation to users' aptitudes and preferences, to the requirements of the tasks to be performed and to the practices favoured in the organization. This increased complexity of the issues of siting computers bespeaks the need for installation management tools for optimizing the design and use of computer tools in an operational context. Evaluation of the user-friendliness of tool-user interfaces is one such instrument.

Evaluating the user-friendliness of an interface (or series of interfaces) means first of all defining the problem of the tool-task-user-organizational dynamics, to derive its main characteristics, from which different evaluation criteria may be defined and built. In other words, the central goal of evaluating the user-friendliness of a tool-user link is to determine users' specific needs and the appropriate elements for enhancing interaction with the tool, to define the most suitable ways of bringing these elements into play. To this end, the evaluation process must be able to draw out information and bring its recommendations into play throughout the process comprising the characteristics of the interaction. This is initiated with the strategic decision to computerize and continues through the design, siting and various development phases of the installed tool. Ultimately, our purpose is to articulate a methodology for evaluating user-friendliness that is integrated into a more comprehensive approach to evaluate the "implementability" of computer tools in the workplace, as an installation management tool.

In this perspective, the goal of this paper is to provide a general description of the issue of the evaluation of user-friendliness and to propose a methodological approach geared to it. It represents preparatory work laying no claim to exhaustive coverage of the topic, but aiming rather to open the door to a longer-term work of reflection and experimentation. In order to derive the main elements of a theoretical definition, the first section offers a brief overview of the characteristics of the concept of user-friendliness. On the basis of analyses found in the literature, the second part specifies the conditions of application of the concept of user-friendliness, first by distinguishing two ways of addressing the tool-user interface, and then defining the tool-task-user-organizational dynamics set as an ingredient of user-friendliness. We will then present the naturalistic investigation as the approach to be favoured in our exercise, following which we will identify evaluation criteria and ways of operationalizing them geared specifically to the issue of user-friendliness. Finally, we will attempt to derive the evaluation methods appropriate to our approach and our definition of the concept.

1. INGREDIENTS OF THE CONCEPT OF USER-FRIENDLINESS OF INTERFACES

Since the early 1980s, technological developments have led to the appearance of a variety of interface peripherals (colour, pointing devices, optical readers, sounds, etc.) which bring with them new opportunities for solving the ergonomic problems encountered by the user. In this context, taking the user into account in designing an interactive computer tool is no longer limited to the development of interaction tools associated with exclusively technical characteristics of the computer tool (keyboard, language, printing details, etc.). It also includes the skills and preferences of the users and different classes of users. As Stevens (1983) points out, the choice of hardware is merely a vehicle for constructing dialogues in an appropriate language, and merely a part (and not even the most important part) of the interface design. Spérandio (1987) also shows that there is no such thing as ergonomic software in itself: adaptation of the object must depend on the characteristics of the users and the tasks they have to perform.

Giroux and Larochelle (1987) are in full agreement with this: the tool-user interface is more than the keyboard, screen, menus, messages, command syntax and semantics, etc. According to these authors (1987, p. 29), it is also the mental model which the user has of the system; and it is the structure of the tasks to be performed, that is, of the more or less optimum strategies to which the user may turn in line with his expertise and goals. One may therefore talk of an **"objective" interface**, that designed by the developer and associated with the system itself, and a user-generated **"subjective" interface**[1] which depends primarily on the user's psychological processes and the context in which the computer interacts.

Thus, in observing the user-friendliness of interfaces, it is the consistency of this tool, and of the relationship among its components, that should be analysed first (Giroux and Larochelle, 1987; Barnard, Hammond, Morton, Long and Card, 1981; Young, 1981). To arrive at a definition of what an optimum interaction would be, one must avoid defining the user-friendliness of a computer tool in terms of its intrinsic hardware or software characteristics. According to Giroux and Larochelle (1987, pp. 35-36), what is important is the impact of these characteristics on the effectiveness of the individual-computer system, which may be measured by different performance criteria. In the final analysis, these are the parameters which should be optimized, since these are the parameters which concretely reflect user-friendliness. In this perspective, the user, the tool and the interfaces enabling them to communicate must not be seen simply as discrete components standing side by side in performance of a task. In fact, the user, the computer tool (including basic hardware, interface equipment and interface

---

1. But the distinction between these two types of interfaces is not an obvious one : one may wonder to what extent the objective interface does not predetermine certain of the user's reactions or psychological processes, attributed here to the subjective interface. If one talks of user-machine interaction, one should also talk of interaction between these two poles (objective and subjective interfaces), and of at least a potential continuity between them.

software), the task to be performed and the conditions for performing this task (ie., the organizational context, constraints associated with organization of work and the specific objectives of the task, involving constraints on the style of interaction to be favoured) are the components of a single set[2].

In this perspective, the concept of user-friendliness cannot be defined merely by the extent to which a tool is easy to use to perform a series of tasks. Nor is it an attribute which emerges in itself from a computer tool. Translating all the aspects influencing a tool's ease of use (or "comfort"), user-friendliness also refers to the organizational conditions which contribute to sharpening users' motivation, so that they will be more or less receptive to technological change (for instance, so that they are willing to devote a relatively substantial effort toward learning and adaptation). What allows one to judge the relative user-friendliness of a computer tool is how it meshes with the interaction linking the **user-computer tool-task-organizational dynamics** poles. In other words, applying the concept of user-friendliness to a computer tool takes on meaning in the way in which this tool fits in with its link with the user, depending on its impact on the execution process and organization of his task, the (formal or informal) recognition of the user's efforts by the organization, and the culture and practices of the organization.

---

2. In the same vein, Eason (1984) talks of the task-user-tool ensemble.

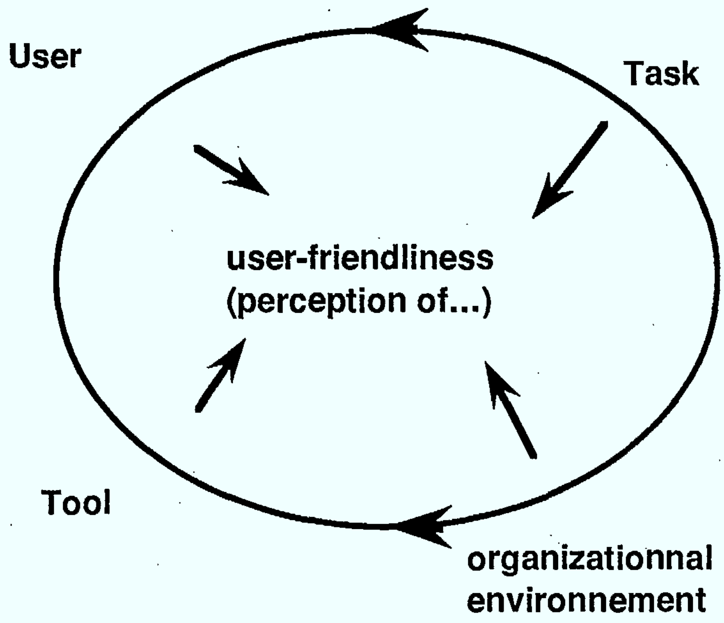

Fig. 1 Ingredients of user-friendliness

The specific needs of a group of users in terms of interface user-friendliness are thus defined in a complex relationship between their backgrounds in terms of technological know-how, the requirements and characteristics of their tasks, and their relationship (perception, expectations and aspirations) with the organization to which they belong. Thus, the question of user-friendliness partly refers to the issue of users' **representations**[3], which are themselves constructed

3. We use here, at a local (micro) level of analysis, the concept of social representation as defines by Denise Jodelet (1988), that is, as a socially shared version of reality. This concept is intended to bring together the mental and social dimensions, to be an interface between the individual and the collective. In fact, social representation refers to the way in wich the individual -- in this case the user -- reacts to a question or situation specific to his environment, thus to the common sense used in the interaction and to socially shared attitudes. Jodelet describes representations in analysis of new technologies as being associated with representation of the task, and with the individual's reaction to the task. With respect to the issues of technological transfer, the author also presents representations as being built through several

from relationships associated with the way in which the tool is integrated with the ways in which tasks are performed and the organization's operational dynamics. In this sense, it must be addressed as an object which is indissociably both social and technical[4].

Seen from this perspective, user-friendliness may not be evaluated by purely quantitative measurements. Instead, it is a quality (in the sense of property determining the nature of an object) which takes its meaning only in relation to the characteristics, aspirations, representations and environment of the users who actualize it through their relationships with the tool. According to Laflamme (1987, p. 43), quality has to do with nuance, detail, the subtle, unique things which make a difference beyond scores obtained on a standardized scale. Quality is what separates or falls between such standardized scores. It is not so much a question of interval versus ordinal scale as a question of meaning. The complex, relative nature of the concept of user-friendliness thus underscores the appropriateness of using an evaluation procedure that integrates analytic tools which are flexible, able to

---

relationships : relationship with the environment, the type of know-how in place (traditional, technological, scientific, etc.), operating logic (relationship with an economic logic, for instance) and characteristics attributed to technologies. Thus, people sharing the same activity (or the same group of activities) in the same environment belong to their own representation universe, from wich the group generates a special universe of standards and knowledge. In our analysis, representation therefore refers to the way in wich user collectively conceive the components and mechanism of their environment, namely the computer tool, the task to be performed and the practices of organization to wich they belong.

4. On the relationship between the social characteristics of organizations and the integration of technological change, see Klein and Hirscheim (1989), Kling and Iacono (1989) and Storrs (1989).

capture the specific features of the area of study and derive detailed, subtly shaded information.

2. CONCEPT OF USER-FRIENDLINESS OF INTERFACES: CONTEXT OF APPLICATION

Before defining the concept of user-friendliness of interfaces any further, the context in which it applies must be specified, along with the empirical elements by which it is defined. To do this, we will first see how the tool-user interface may itself be designed. The literature on the subject uses different approaches in relation to which we shall attempt to place ourselves. We shall see how some authors present their definition of user-friendliness as relative to the features of the context of the interaction, including the characteristics of the tool, the features of the users and the tasks to be performed, and the specific nature of the organizational context.

2.1 Tool-user interface: two approaches

Norman (1983) presents a series of principles underlying the approach to the user-computer interface, from the perspective of the study of the human factors associated with operating computer tools, presented as an individual field of scientific practice. To this end, he favours taking into consideration the characteristics (powers and weaknesses) of the technology itself in the study of phenomena he describes as **parallel** or **independent** (work environment, etc.). This involves establishing at the design stage (and not after the event) a close relationship with the various local constraints (technical, organizational, budget- and time-related, etc.) surrounding siting of the tool and, above all, introducing appropriate design principles. The idea is to build an interface whose parameters can be easily modified, independently of the tool, thus offering the possibility of creating personalized interfaces. The goal is to allow greater flexibility, both to the designer and to the interface itself.

Several authors who espouse the human factors approach (including Bennett, 1986) or the cognitive ergonomics approach (like Vanneste, 1987) follow the same lines as Norman, stressing the importance of encouraging builders to separate the design of the presentation and dialogue with the user from the design of the content of specific applications. In Bennett's view, keeping the interface function separate from the rest of the tool must be encouraged, at both the design and the evaluation stage. For Vanneste (p. 53), it is up to the evaluator (or ergonomist) to acquire modelling and prototyping tools for developing and validating the specifications of user-computer dialogue interfaces alone. The author thus proposes to isolate the interface from the "internal" and "purely technical" aspects to facilitate their adjustment and flexibility.

But several authors challenge this approach of separating the user interface. Graham Storrs (1989) makes the distinction between **interaction** and **interface**. An interaction is an exchange of information by which two agents each modify the state of the other (i.e., the composite of the states of an agent's beliefs, goals, plans, intentions, attitudes, values, dispositions and physical state), while an interface refers strictly to the information channels which support and influence (through their characteristics and the constraints they present) the content and structure of the exchange. In this perspective, the user interface is not something which can be treated separately from the rest of the system, inasmuch as there is no clearly defensible place to draw the line between front-end and back-end of the system.

Defining the parameters of an applied psychology of the user, Moran (1983) adopts a similar position: "Psychologically, the user interface is any part of the computer system that the user comes in contact with." (p. 5) According to the author, it is thus impossible to differentiate explicitly the interface elements from the overall internal structures of the computer tool, since the latter are also likely to influence the user's perception or representation of the tool. This "user's conceptual model"[1] is an integral part of the user interface. For Moran, the issue of creating the parameters of the user-task-machine dialogue is to integrate three ways of approaching the interface question: that of the technician, for whom real progress involves increasing the machine's technical capabilities; that of the designer, who, based on his own intuition, must predict the approach that will be

---

1. Moran presents the psychology of the computer use as a subfield of computer science, wich he defines as an extention of human factors engineering, likened to European ergonomics. The author prefers to adopt a more comprehensive perspective by talking of «applied psychology», wich he approaches from the viewpoint of the psychology of the user as individual (rather than social agent). He focuses on the cognitive aspects of the user, wich include learning, performing, and reasoning (and not motivational, emotional or personality aspects of the user). His approach is based essentially on empirical research.

the most user-friendly for the user; and that of the psychologist, whose role is to ensure "reliably", on the basis of objective studies, that a satisfactory tool-user interaction results. R.M. Young (1981) reaches similar conclusions in his analysis of representations resulting from the use of different pocket calculator models by users with no in-depth technical knowledge of these devices.

The human factors approach could be called **interventionist** to the extent that it essentially aims to facilitate and accelerate procedures for evaluating and modifying tools that are on the point of being installed. Moran instead promotes a **global approach** to the tool-user interface whose objective is essentially to generate a better overall understanding of the psychology of the user. How these approaches are defined thus largely depends on the intervention method preferred in the area. If the goal of the evaluation is to infuse the installation process with the viewpoints, needs and expectations of the users, thus taking into account their perceptions rather than standard, pre-established criteria, it will gain by considering a global interface approach[2]. This way of conceptualizing the tool-user interface seems to us to be more closely geared to the need to take into consideration the complexity and variability of the tool-user relationship.

---

2. But this must not imply an exclusive choice inasmuch as it may be necessary to validate field study results by «triangulation» with controlled environment studies, more closely resembling Norman and Bennett's approach, for instance.

The aim here is not to restate Moran's position, which opposes the psychology of the user as an individual to his consideration as a social agent. It is important first of all to establish the level of knowledge which the user needs to feel comfortable with the tool in performing his task. Then it must be confirmed whether this knowledge (or lack of knowledge) actually influences his representation of his task and work tools. What we propose is to integrate a global approach with the evaluation of the user-friendliness of interfaces, presented as arising out of the tool-task-user-organizational dynamics set, as an entity that is both social, technical, and cognitive. If the interface is not limited only to the user's links with the system commands and language procedures, the evaluation must also be able to capture the ramifications of these links in the user's cognition, their actualization in an organizational dynamics and a specific user environment.

2.2 The tool-task-user-organizational dynamics set as an ingredient of user-friendliness

In a substantial portion of the literature, the concept of user-friendliness implicitly refers to a computer tool-user interface geared to users' needs, in terms of ease and comfort of use, ease of learning, suitability of functions and attractive presentation. Nonetheless, the behaviour of the user **changes over time** and is **complex** (Moran, 1983; Norman, 1983; Stevens, 1983; Eason, 1984). Thus, tool-user interaction (including interface design) must be able to capture this complexity. In this perspective, Stevens (1983) demonstrates that interface design must adopt an approach encompassing broad, flexible criteria for defining its objectives, in order to cater to diversified needs, expectations and satisfaction criteria that are variable over time. Starting from this point, the author demonstrates the appropriateness of categorizing user types, according to the task to be performed and the knowledge acquired with respect to computer tool use (distinction between novice and expert).

A first dimension of Stevens's analysis is based on evaluation of **users'** knowledge, particularly in order to distinguish their specific needs, depending on their level and type of expertise. This distinction takes into account users' distinctive behaviours, depending, for instance, on their specific background in terms of courses and professional expertise, their representation of their work, their motivations and their concerns vis-a-vis installation. The user's motivation depends upon the degree of mesh between his aspirations and the characteristics of the computer tool. Moreover, these aspirations change over time, as his use and knowledge of the tool evolve and change. Faced with the difficulty for the same tool to be equally user-friendly for various users, Stevens proposes placing greater weight on the user's identity, in order to establish use models for the tool. A second dimension refers to the structure of the task, that is, to the type of task and the level of skill it requires. In other words, one must know to what extent the task is varied or repetitive, whether it involves frequent or intermittent use of the tool, thorough or partial technical knowledge (eg., programming or word processing), etc.

Norman (1983) takes a similar approach. The author emphasizes that, owing to the variability of needs, the specific structure of the user's activities (main task and subtasks) must be understood, whence the importance of linking the operations of the **tool** consistently with the **structure of the task** to be performed. But the latter must itself be considered jointly with the individual features of the user, whom Spérandio (1987) describes as being above all defined by his relationship to the task he performs. Among the features characterizing the user, Spérandio notes in particular his knowledge and know-how relative to the task, workstation and profession, and above all the reasons (motivations) for which he operates. (pp. 17-18) It is therefore through the close link between the **user**, his conception of his **task** and the functional characteristics of the **tool** that the reasons for using computer tools may be seen. Christol (1987) mentions the need to also refer to the existence of different work environments and variable circumstances (brief excess workload, incidents, etc.), and thus to the impact of a specific **organizational context**.

These observations present the processes making up tool-user interaction as necessary indicators for interface design. They are every bit as necessary for designing evaluation of their user-friendliness. The same type of computer tool may have to meet a vast range of expectations and needs from various classes of users, who convey different representations of their work and operate in different contexts. The design of an interface that is intended to be user-friendly therefore faces varied requirements that will evolve over time. In this sense, the investment in learning and use which may be demanded of users, while providing a level of ease of use which they find satisfactory, may be determined only from the background (knowledge, work and use experience, etc.) of the users and the specific tasks performed in specific contexts (Tyler and Treu, 1989). Therefore, a priori, no interface is user-friendly in itself. Only through the local characteristics of an installation site may one define what is pleasant, difficult or too simple (so not sufficiently motivating for an experienced user) to use, what fits in well or poorly with a pre-existing work process, what is perceived as threatening or otherwise.

The criteria for establishing whether a technological tool is user-friendly may be established only in relation to the dynamics into which the tool fits and in which it participates, with respect to users' specific goals and a specific organizational environment. From a methodological viewpoint, the complexity of this reality precludes development of standardized, rigid evaluation grids. If the analytic criteria for establishing these categories may be generalized, their application and the data they reveal, if they are to be opportune, must be defined in terms of the characteristics specific to each evaluation site. In this context, the role of the evaluation is to identify on-site what is needed for a tool to meet users' expectations in terms of performance and ease of use. In addition, this tool must offer sufficient flexibility to follow the evolution of users' skills and needs.

3. A NATURALISTIC APPROACH TO ASSESSING USER-FRIENDLINESS OF INTERFACES

The multiplicity and complexity of tool-task-user-organizational dynamics relationships mean that the user-friendliness of a computer tool cannot be defined solely in light of standardized analytic models. If the characteristics of the user-friendliness of a computer tool are defined only in relation to a potential variety of needs, purposes, representations, knowledge, practices and relationships to a socio-organizational context, its evaluation must capture these local features and their complexity. Assessing user-friendliness therefore necessarily depends upon identifying the characteristics and needs which emerge at the installation site, whence the need for a flexible, "reactive" approach. In this sense, such a process gains from a **naturalistic** approach, that is, an approach based on a systematic relationship to site characteristics, opinions, perceptions and representations of the users and the players they come into contact with (managers, other work groups, etc.).

Along similar lines, Patton (1983) proposes an "a-modellist approach" to evaluation, which, through its flexibility, would provide for explicit recognition of the diversity of the methodological options to which evaluators may turn, depending on the specific needs of the site under study. The naturalistic approach aims to meet these conditions.

It is similar to an "anthropological-type holistic-inductive model"[1] (Laflamme, 1987, p. 22) which aims to understand social phenomena rather than predict them (or at least to understand before predicting). Guba and Lincoln (1983, 1985) put forward naturalistic investigation as being characterized by the consideration of the subjects' values, as well as its refusal to claim total objectivity and its imperviousness with respect to these values. One of the premises of naturalistic investigation is that the phenomena studied acquire their meaning not only in themselves, but also from their contexts. Naturalistic evaluation therefore fits in with an installation approach that could be called "participatory", aiming to consider the opinions and ideas of the local players (users, managers, etc.).

A naturalistic-type evaluation approach must make the investigator-respondent relationship recognizable, since it considers this as a source of data in itself. In this perspective, the evaluator will show flexibility in applying his evaluation methods and will avoid being guided by his own certainties: nothing is given beyond what the data confirm. Thus, the traditional criteria of validity will be replaced by criteria of "credibility", "transferability", "dependability" (of results in relation to their data and context) and "confirmability". Thus, research

---

1. That is, a model wich studies phenomena on the basis of the human relationships taking place there, through an approach wich, conceiving the components of a given situation as a whole, develops general conclusions through reasonning based on the articulation of facts as they emerge from the site under study. This approach is opposed to the classic, deductive scientific approach, wich seeks to compile and analyse facts on the basis of previously established premises or hypotheses.

must be conducted on the basis of the specific "nature" of the evaluation issue, meaning that the naturalistic paradigm must be **emergent** (stemming from data and researcher-site interaction) rather than preordained (deterministic). In this sense, naturalistic evaluation favours placing the object of study in a grounded theory perspective, where the theory-field link enables agents and data -- rather than just the evaluator -- to express or translate a viewpoint.

For Guba and Lincoln, the naturalistic paradigm is based upon a series of axioms, described as non-absolute, that may be selected, replaced or modified according to the needs of their context of application. These axioms are as follows:

(1) Realities are multiple, constructed and evolutionary; the reality observed in an investigation may thus be validly understood and interpreted in several ways (ontological dimension).

(2) The investigator and the object of his investigation are inseparable: they interact and influence each other mutually (epistemological dimension, or scientist-knowledge relationship).

(3) Only working hypotheses tied to the time and space (context) of the investigation are possible. Thus, the naturalistic investigation refuses to incorporate into "laws" the statements stemming from data, which apply only to their specific context of emergence. But, as the authors point out, prior experience and research may be used

as references for partial generalization of an approach, method, or way of addressing a certain type of study site (generalization).

(4) Cause-effect relationships (or causal chain): all entities take shape mutually and simultaneously. Consequently, it is impossible to establish causes and separate them from their effects.

(5) Role of values in the investigation (axiology): an evaluation is loaded with values, through the investigator's choices, focus and approach, his position with respect to the entire research field, and the context of this approach. In this perspective, the evaluation procedure is based on a consistent framework made up of objectives, representations of reality and subjective views. The evaluator must be aware of this and ensure consistency with his approach and the realities of the object of his activity. Hence the importance of good communication between the evaluator and the subjects and sponsors of the evaluation, and of sufficient broadmindedness for mutual exchanges of viewpoints.

To summarize the main elements characterizing a naturalistic evaluation approach, there is the importance of the evaluator's **adaptability**: his activity is based on the one hand on tacit knowledge, used as "propositional knowledge", and on the other hand on "human tools" analytic methods (interviews, observations, documentary analyses, etc.). Field analyses are **inductive** (emerging from data) rather than deductive (based on pre-established hypotheses).

Deliberate samples are preferred to strict proportional representativeness, which tends to eliminate subjects that are "marginal" as to number, but whose impact may be of the greatest importance. Moreover, the decision to conduct the investigation must be taken only after ensuring that the key players (formal and informal) at the study site will collaborate. Consequently, the researcher must acquire prior knowledge of the site and negotiate the parameters of the evaluation. Finally, interpretation of data requires the participation of respondents-players, who are invited to participate in negotiating what is worth retaining and in interpreting the data.

In this perspective, the evaluator opting for a naturalistic-type approach conceives the study of phenomena in an inductive manner, as they appear to him, avoiding rebuilding them through external controls or manipulations. The representations and subjective views of the players on hand at his study site are seen not as subjective incursions into an objective scientific approach, but as indicators and components of a specific reality, which is partially the outcome and partially the generator of these representations. Thus, evaluation of the user-friendliness of a computer tool must define its parameters in relation to the specific context in which installation takes place. Since the identification of needs and definition of the criteria of user-friendliness of a computer tool depend on locally defined considerations, it is only in relation to the latter that the evaluator can establish a suitable analytic process that is likely to contribute to successful installation.

Taking into consideration users' positions and their socio-organizational dynamics favours the use of qualitative methods of gathering and analysing information, based on the site study using subject-evaluator dialogue techniques, semi-guided interviews, open-type questionnaires, control groups, different observation techniques (modelling with logs, participatory observation), documentary research, etc. But this does not exclude using quantitative methodologies when the site lends itself to this (for instance, to highlight major trends in an organization, in order to indicate paths to be explored or questions to be raised with players). What a naturalistic approach recognizes is the need to select evaluation methods and tools first of all in light of the specific characteristics of the site, rather than using "standardized" grids.

Faced with the complexity and variability of the characteristics of user-friendliness, the evaluation may be built only from several methodologies and several data-gathering sources. To this end, we suggest placing the entire process in line with a **formative**-type evaluation (as compared with an impact study approach, for instance), focusing on the study of the overall process of developing and installing the tools, "step-by-step" monitoring of its evolution and harnessing of the findings of the evaluation as they appear. This translates first into placing the project in a context. A picture of the organization as an installation environment is drawn, highlighting the main features of its historical evolution, the socio-dynamic characteristics occurring there, the objectives of change and the perception which the players on hand have of this. At the same time, it is important to be familiar with the users (their academic background and professional culture) and the tasks to be performed. Finally, establishing user-friendliness criteria involves placing these factors in relationship with the identification of the type of technology introduced, its functions, its operating modes and the leeway offered by its level of flexibility.

4. CONCEPT OF USER-FRIENDLINESS: OPERATIONALIZATION AND ANALYTIC CATEGORIES

4.1 Analytic categories

A brief overview of the criteria for evaluating and identifying user-friendliness most often used in the literature will enable us to better understand how it fits in with the tool-task-user-organizational dynamics relationship, as well as the latter's relativity, variability and complexity. The analytic dimensions most often used in literature as being usable in defining the user-friendliness of user-machine interaction may be grouped in four main categories: (1) the (intrinsic) **suitability** of the tool (which includes its functionality, flexibility, compatibility of its functions with task performance procedures and consistency of its operation with users' needs and social representations, etc.); (2) its **contribution to task performance** (i.e., its contribution to facilitating and enriching the task, the quality of its inclusion in the pre-existing task structure); (3) the **conditions of learning and appropriation** of the tool (which underlies the availability for users of the necessary training and information resources, incorporation into the tool of error prevention and error recovery assistance functions, etc.); and (4) the **"usability"** of the tool (relationship between potential and actual use).

Far from being unchanging over time, dependent on the context of use, these analytic dimensions are intimately linked with each other. They stem from a series of choices and tradeoffs between, for instance, use needs and technical and financial constraints, which -- potentially -- occur throughout the tool's use from the moment of its design to the moment when it is deemed necessary to replace it, via all its modifications. But first let us look, for each of these four categories, at the definition features suggested in the literature, as well as examples of analytic dimensions of user-friendliness which we can derive from them.

4.1.1 Suitability of the tool

(a) Conceptual definition

The functionality and flexibility of a computer tool are two factors intimately linked to the needs and constraints of the task to be automated and the target user category. Spérandio (1987) explains that the variety of users, tasks and needs with respect to the same type of technology demonstrates the need for an analysis of the work process into which the tool is to be integrated, as a prerequisite for its design. He describes it as the role of a schedule of conditions to determine the direction to be taken in defining the functions, coding and modes of dialogue[1], depending on distinctions between novice, occasional or expert users, and between rare, frequent or repetitive tasks. The author notes a variability of compatibility criteria among (graphic and other) representations generated by the machine and the mental representations which users have of the installation, depending on the class of users, their level of technical culture, and their position in the work organization and their individual (psychological) characteristics.

Moran (1983) defines the user-friendliness of a computer tool first in terms of the user's activities. To understand the latter's needs, one must therefore be able to understand the goals he has to accomplish, depending on the constraints imposed upon him by the structure of his task, what he knows and his own mental information-processing limits. The definition of an optimum behaviour (or series of behaviours) is generally the outcome of tradeoffs between different dimensions, with respect to priorities arising from each particular situation. The concept of **acceptability**, as defined by Moran, refers to **subjective** evaluation of the tool by the user, for the performance of a series of tasks. An important factor that may lead to a positive evaluation by users lies in the quality of the inclusion of the tool's processes in pre-existing task and work habits. To describe the conditions of this inclusion, Stevens (1983) talks of the naturalness of the interface. The «naturalness» of the hardware part of the interface translates into the

---

1. For instance, the vocabulary used in dialogues, abbreviations or symbols must respect previous habits, remain consistent within a given application, and among applications concerning the same operator. (Spérandio, 1987, p. 18)

use of a combination of processes and procedures which approximate those normally used by the user, depending on his habits and work methods, his particular academic background, etc. The result is a broadening of the computer language structure, which goes beyond the mere command language. Of course, the idea is not to reproduce human communication, but to make communication with the machine more pleasant. Creating an interface in an evolutionary perspective, for instance, allows it to remain user-friendly with respect to development of the user's skills. To achieve this result, the interface should, for instance, be adaptable by the user himself, as his requirements evolve.

In this perspective, Tyler and Treu (1989) propose an adaptive interface design based on the "modularity" of the interface system, which resides in a separate module that may be easily modified independently of the rest of the programming. The user is thus offered the opportunity to organize and modify the content and presentation of the assistance with which the interface can provide him (error messages and operating instructions), in a personalized way, depending on the type of use of the tool with which he is the most comfortable. The philosophy underlying this interactive interface design is to promote learnability rather than user-friendliness, to encourage non-expert users to learn and give them the opportunity to evolve in their use. Here, user-friendliness is seen in terms of adaptability and need to respond to a multitude of needs[2].

---

2. The two authors are not, however, very explicit on the way of defining the range of these needs, evaluating the actual learnability of the interface, and

The question of adaptability leads us to the concepts of **consistency** and **compatibility** (Barnard, Hammond, Morton, Long and Clark, 1981; Giroux and Larochelle, 1987), developed in cognitive ergonomics and applied psychology. The purpose of using these notions is to contribute to a software design consistent with users' cognitive referents, their representations of the operation and "role" of the tool in performing tasks. Consistency designates the interface's internal relationships, which refer to the type of knowledge or representation to which the user refers to carry out his task, to determine the best way of executing an operation, etc. In concrete terms, consistency may translate into using, in the different software to which a user may turn, a similar presentation, logic and language to operationalize commands. Compatibility on the other hand refers to relationships between different "blocks" of knowledge and representations. For instance, the terms designating commands, or the arrangement of the commands to be given to execute an operation, may be incompatible with the representations conveyed in everyday language.

The analytic framework we have just described clearly illustrates the variability and relativity of the criteria for defining the concept of user-friendliness. If, like Spérandio, one identifies design principles which may confer a user-friendly character on the software at the very time of its design[3], it is essential to bear in mind that such a

---

users' predisposition to investing the long-term learning effort necessary to effectively broaden their technological knowledge.

3. Spérandio suggests in particular minimizing repetitive actions, as well as the waiting times not specified to the user; minimizing the use of non-significant

"preventive" evaluation of the intrinsic characteristics of the tool will not be complete unless it is followed up by a formative evaluation of its **implementability** in an operational context. The goal of this second stage in the evaluation process is to confront the characteristics of the tool with the local features of its context of use.

(b) Some analytic dimensions

Suitability of functions:

Essentially at issue here is verifying the usefulness of the tool's functions and capabilities, in line with users' needs and the particular requirements of their tasks. Observations gathered at this level may make it possible to better adapt the tool's functions to the tasks to be performed.

Flexibility:

Evaluating the flexibility of a computer tool means seeing, on the one hand, whether it offers sufficient opportunities for progression, depending on the amount of information -- help functions, etc. -- which the user needs, or the type of task to be performed. On the other hand, it means exploring the tools' opportunities for modification and

---

codes; ensuring that the user can always take back control of the dialogue or stop an operation in progress; explaining the impacts of the action performed (or the commands given by the user); protecting the user against destructive actions he may make unintentionally, by allowing him to cancel previous actions and return to the previous stage; and making sure that menus and trees are not too long.

adaptation (such as adding or replacing components) in line with users' needs and the foreseeable evolution of these needs.

4.1.2 Contribution of the tool to task performance

(a) Conceptual definition

The user-friendliness of a computer tool may be partially evaluated from its contribution to the speed of task performance and simplification of that task. But the reality is more complex. Proposing a review of each of the main elements by which user-friendliness is defined, Stevens (1983) criticizes a large portion of the literature on the user-friendliness of interfaces for assimilating the concept of "simplicity" to that of "user-friendliness". He recognizes that this is partly justified. Cognitive psychology has contributed to demonstrating the extent of the human ability to perceive, with man interpreting and understanding complex messages from simplified representations (symbols, for example.)[4] In a well-arranged design, a large amount of information may thus be assimilated from little data. Stevens does, however, qualify these conclusions by pointing out that people have no objection to coming to terms with complexity, provided they have the means to master it, and provided this complex situation is structured in accordance with their universe of reference.

---

4. Stevens refers here to the work of Myers, Nelson and Smith.

The designer must take into account the composition and operation of the user's universe of reference. But, Stevens asks, from the moment software design affects a large number of users, how can one respond to these various classes of users, with varied expectations, who have to perform different tasks? If "everyone is equal" when faced with a simple dialogue, Stevens observes that they will not always be equally satisfied, though, at least if one agrees to take into consideration the variety and changing character over time of users' needs and motivation criteria. In this perspective, Stevens tells us, the designer may, aside from developing technical abilities, increase the usefulness of the tool for the user in three ways: (1) by giving it an obliging "personality" in order to make it pleasant to use; (2) by ensuring that the tool provides the user with sufficient assistance (access on demand to information about operations); and (3) by providing error messages which, for users with little experience, explain the nature of the error and propose means of correcting it. The match between the content of messages and the class of user thus gains by being considered.

As we saw earlier, it is generally impossible to conceive as homogeneous the user groups who will have to use a computer tool, as well as the tasks they will have to perform. A user-friendly tool will have to fit appropriately into a task structure, a set of practices, socio-organizational relationships, representations and knowledge. It must meet a series of needs specific to the work site and the organizational context where it is installed. It is by evaluating the qualities possessed by the tool to meet these needs that it may be established whether its contribution to task performance of a specific user group suits the latter's needs and expectations in terms of user-friendliness.

(b) Some analytic dimensions

Performance:

The aim of evaluating performance is to verify the contribution of the tool to task performance. Thus, one may check whether the tool provides significant time savings, whether it offers the services users expect of it, and whether the time and effort it saves in the main task justify the learning effort and additional technical tasks it may impose.

"Comfort" of use:

The evaluation of ease of use applies to the specific characteristics of the tool, such as execution of certain operations, constraints on execution of the task which the tool may present, etc. The aim of using this criterion is to identify "irritant" effects which the tool may have on users. This concerns, for instance, time required to execute a task, complexity of the various operations, etc.

4.1.3 Conditions of learning and appropriation of the tool

(a) Conceptual definition

To a large extent, a user-friendly interface is expected to be conducive to speed and ease of learning, with respect to the context of use, the power of the software, and differentiations between the acquired knowledge and the knowledge actually used (Giroux and Larochelle, 1987). User-friendliness also concerns the retention of learning (depending on frequency of use, ease of identification of interface codes, etc.) and the transferability of this learning (Young, 1981). A partial definition would therefore consist in saying that a user-friendly interface requires minimal effort from the user in terms of training and adjustment, that it requires basic knowledge, shared by the greatest number.

Nonetheless, training[5] itself is a tool providing access to mastery of a certain level of complexity, allowing user-friendliness criteria to evolve by modifying users' levels of competence and their technical or professional culture. In fact, training influences the range of opportunities for application of tools and organizational modes: choices and focuses in this area comfort or eliminate knowledge, and alter power relationships (Christol, 1987, p. 13). The suitability and

5. By training, we mean, in addition to formal education, on-the-job training, dissemination of documentation and information on the use and operation of the tool, etc.

complementary nature of the measures accompanying installation of the tool (training and information dissemination methods and procedures) are therefore decisive for the overall implementation. In this perspective, evaluation of "user-friendliness needs" is partly tied to the relationship of goals set with respect to the scope of the modifications desired, users' previous knowledge and the time and resources which the organization is prepared to supply to develop or adapt this knowledge.

Moreover, for Stevens (1983), effective, "comfortable" use of a computer tool does not necessarily depend on the user's overall level of comprehension, but on his mastery of the knowledge that can be used in his work. In other words, the user does not need to know all the characteristics of the tool's operation to be able to perform his task and achieve his particular objectives. The required level of understanding of the operation and opportunities available thus varies according to the requirements of the task and the users' field of expertise. This variability means, according to Stevens, that it is difficult, at the design stage, to predict at what point the interface must be transparent if one is not familiar with potential users and the context in which the tool will be installed. The procedure thus involves an approach whereby local needs are anticipated. Implicitly, such an approach recognizes the need to opt for a procedure which can capture the diversity, complexity and relativity of the ingredients of user-friendliness.

(b) Some analytic dimensions

Feedback provided by the tool:

Feedback concerns instructions and information transmitted in the course of operation by the tool. The idea is therefore to ensure that the tool is able to provide help and information functions that are sufficiently complete and easily accessible, that the menus and windowing are organized clearly enough for users. But the situation must also be avoided where feedback elements needlessly weigh down operations or inconvenience users by flooding them with unnecessary information.

Training and information:

It is important to bear in mind that needs in terms of user-friendliness, as well as the ability of the tool to meet all the other evaluation criteria, largely depend on the effort invested in user training and dissemination of information (including technical support to users). First, one must see whether the content and format (speed, duration, teaching approach) adequately meet users' expectations and specific needs and whether they apply properly to work situations.

Moreover, the scope of resources to be devoted to technical support and assistance will determine the extent of the technical knowledge which users will require. In this sense, training needs depend not only on the gap between the procedures for using the tool and the user's knowledge, but also on the level of autonomy required by users to operate, and on the assistance and support functions provided by the tool itself.

4.1.4 Usability of the tool

(a) Conceptual definition

Acceptance of a computer tool does not necessarily depend on its potential use in performing given tasks, but also on the applicability of characteristics that may be useful to the user, in relation to his socio-cognitive context. This complex problem may be identified as the challenge of "usability" (Eason, 1984; Goodwin, 1987), as a limitation according to which potential usefulness may be translated into concrete practices (or actual usefulness). Usability, as defined by Eason, is therefore a means of recognizing the discrepancy between the **functionality** or potential use of a tool and the **level of use** which unskilled users may or wish to attain. The goal of usability is to determine difficulties which hamper optimum use of functions that users find useful. This gap between potential and actual use may be caused by difficulty operating the tool, but also by constraints associated with the conditions of use, such as the specific requirements of the task or users, time available for learning, etc. In an operational context, the symptoms of a usability problem are revealed by under-operation of the tool, by what Eason calls "non-use", or by "remote use" (one has others use it).

The design of a usable computer tool, according to Goodwin, thus requires that one take into consideration the characteristics of future users, their knowledge of computing, the time anticipated for use of the tool and evolution of their needs as they acquire expertise. On this topic, Goodwin identifies three types of factors which give the tool its usability. First, factors such as training, accessibility of terminals, and modes of operation characterizing the "culture" specific to a work site are **organizational** in nature. Then, usability also depends on certain **hardware** characteristics, such as response time, etc. Other factors, finally, are more directly associated with **software** design, that is, the potential and characteristics of its functions in relation to the requirements of the task, screen design, command language, user-machine dynamics, etc. One also has to take into account the evolution over time of the task, needs and requirements of users.

At a more conceptual level, Eason presents usability as stemming from the interaction between three variables: tool, task and user. The interaction of the various components and the various social players required by installing the computer tool (hardware, software, manuals and documentation, users, managers, trainers and "local experts"[6]

6. Eason identifies local experts as users who, owing to a special interest, skill or technical knowledge, have developed a mastery of the tool greater than the average of their colleagues. Informally, they often act as advisors or monitors to the users in their immediate entourage. Noting the very positive impact of the presence of local experts in a work unit, Eason recommends that managers recognize and encourage their work of support and emulation vis-a-vis other users.

(means that the tool must be designed as a **socio-technical** entity[7]. The task and user variables are the originators of constraints and tradeoffs between the requirements of the task and those of the tool, between users' various needs (depending on whether they are novices or experts, for instance). These variables are built from the reaction of the user to each portion of the task.

Placed together, the tool, task and user variables produce the context on which the user bases himself to "construct" his judgment as to the best way to set about carrying out a specific task. This is what Eason calls an "implicit cost-benefit" assessment by the user, with costs being calculated in terms of effort (primarily in seeking new practices and new ways of developing new knowledge) and benefits in terms of skills and facilitation of specific tasks via the system. From this series of cost-benefit assessments will emerge, depending on users' motivation, a strategy of use. Within the framework of the procedure for evaluating user-friendliness proposed here, taking into consideration the "strategies of use" (or non-use) contributes to defining users' positions with respect to the method and context of performance of their tasks. Ultimately, this refers to users' representations of the suitability of the tool, which includes the "value" assigned to the tasks which it enables them to achieve, and the position in the organization which it may allow them to acquire, improve or

7. In other words, the series of elements comprising the tool (functions, choices, decisions, tradeoffs vis-a-vis constraints of various types, etc.), as well as the impacts it generates (on the work organization, on relationships in the organization, etc.), are indissociably both social and technical in nature.

maintain. In this sense, these strategies, by illustrating the possible discrepancy between the potential use and actual use of the tool, are a product of the uniqueness of the tool-task-user-organizational dynamics relationship.

(b) Some analytic dimensions

Ease of use:

There are not really any standard characteristics by which it may be established in advance, with any certainty, to what extent a computer tool will be easy to use for its potential users overall. These criteria must be defined on the basis of the technical knowledge of users at the time of installation, depending on the scope of the training program and the technical support offered, the type of task to be performed (ie., will the tool make the task performance procedure simpler or more complex, etc.).

Ergonomic factors:

Ergonomic factors concern evaluation of the layout of the site and the physical effects associated with the way equipment is used (frequency, duration, etc.). Inadequate workspace, eye fatigue and problems entailed by sitting for too long are examples of ergonomic factors which may detract, in the medium or long term, both from their user satisfaction and from the performance of a computer tool.

Table 1 Analytic categories and dimensions

Suitability of tool	Contribution to task performance	Learnability and appropriation	Usability
Suitability of functions	Performance	Feedback from tool	Ease of use
Flexibility	«Comfort» of use	Training and information	Ergonomic factors

4.2 Operationalization: dynamics of compromise and tradeoffs

As we mentioned above, the categories for analysis of user-friendliness are interdependent and interact with each other. Their relevance and respective importance largely depend on the characteristics, needs and constraints (time- or budget-related, organizational, etc.) specific to each context of installation. The interaction and sometimes antagonistic relationships between the requirements of the different criteria for defining user-friendliness present design choices as stemming from one or more tradeoffs (Giroux and Larochelle, 1987). The installation process itself is a dynamics of relatively formal compromise and tradeoffs, involving both social and technical factors.

Along the same lines, Bennett (1986) describes the installation process as marked by the omnipresence of compromise and tradeoff between two categories of objectives: the interface designers' and the users'. To this end, he proposes an evaluation procedure built from four main indicators. First, **flexibility** of the interface with respect to the tool, which refers to the possibilities, within the same category of environment, of adapting the tool to new styles of interactions. Then **learnability**, whereby the level of performance attained by an adequate proportion of a sample of potential users is determined, following a certain period of training. **Quality control** of the finished product, which involves verifying whether the task may be carried out by the subject sample at a given speed, with less than a pre-established error rate. Finally, evaluation of users' **attitudes** by finding out from the latter, after they have tried out the tool, whether they wish to carry on using it, whether they feel it allows them to develop the means to enhance their productivity, etc.

These compromises and tradeoffs are defined in terms of the choices offered by the technologies used, according to user categories, the objectives of the design, and the possibilities of adapting the interface in line with the choices made. The tradeoff analysis proposed by Bennett is an example of a quantitative human-machine interface design tool. It is intended to facilitate technological choices which interface designers have to make. The evaluation of user-friendliness, inasmuch as it fits into an installation process which takes account of users' needs, expectations and purposes, may partially repeat the tradeoff analysis by means of tools for gathering qualitative data on the requirements and impact of the social relationships surrounding the introduction of technological change in an organizational dynamics.

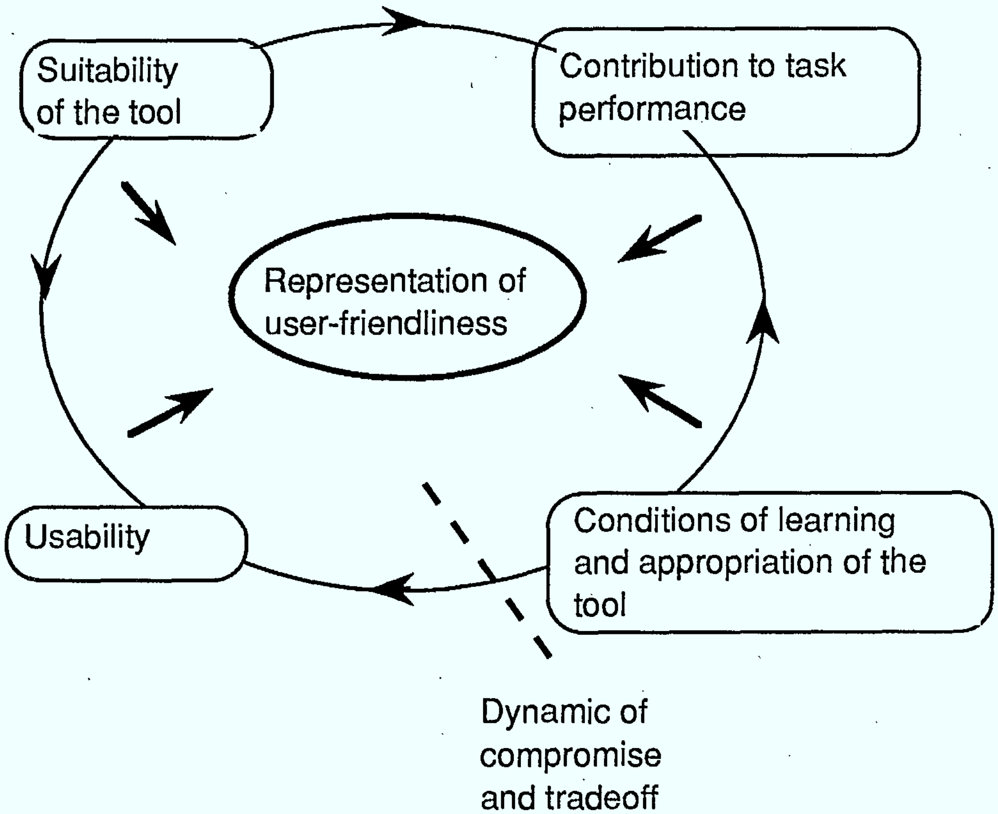

Fig. 2 Categories for analysis of user-friendliness

4.3 Users' "subjective" opinions as a tool for defining and validating evaluation criteria

The main themes for user-friendliness evaluation raised in the literature, namely users' needs and purposes, the functionality and characteristics of the tool, and its relationship to the organizational dynamics, may be described on the basis of data around questions of several types. The analytic dimensions raised are potentially interrelated, and even interdependent. Moreover, this overview enables us to observe that there are not really any standard indicators for predicting with any certainty to what extent a computer tool will be user-friendly for all its potential users and whether it will meet their specific needs.

On the basis of the field study (interviews, observation, control groups, etc.), some or all of the characteristics of the installation process may point to corrections to be made to the tool or to the procedure for implementing the technological change. The definition of user-friendliness criteria is based first of all upon the **satisfaction** of users and managers with regard to the different facets of use of the tool, and thus on the concept of **acceptability** of technological change.

The concept of acceptability refers to the subjective evaluation of the tool by the user, for the performance of a series of tasks. In concrete terms, such indicators as users' satisfaction level and perception of the tool make it possible to establish the most satisfactory tradeoffs with respect to the changes involved in using the tool and the practices it is wished to perpetuate. The concept of acceptability also refers to identifying the apprehensions of users or other groups in the organization with respect to anticipated effects on the task, qualifications, performance criteria, etc.

In this context, the definition of criteria for evaluating user-friendliness gains by using, first of all, the way in which users themselves subjectively judge the different aspects surrounding installation and use of the tool. This approach allows the evaluator to bring out the variety of the information needed to establish indicators of the tool's user-friendliness. It is therefore in light of field data that one may decide which indicators of user-friendliness are relevant, whether it is appropriate to consider other analytic dimensions, and how the evaluation criteria stemming from them must be specified.

5. EVALUATION METHODS

Several methodological tools and procedures are proposed in the literature for evaluating the different aspects of the user-friendliness of computer tools in relation to the requirements of the task and users' needs. Without claiming exhaustive coverage, we will describe some of these methods which may apply, in all or in part, within the framework of what we have called a "naturalistic" evaluation approach. First, we will examine some proposals from French software ergonomics, which has developed several analytic tools primarily intended to sustain software design. Then we will see how different authors, identifying with applied psychology, cognitive ergonomics or the human factors engineering approach, treat two evaluation procedures which seem to us to be complementary: controlled environment evaluation and field studies.

5.1 Some methods for evaluating needs

Mazoyer and Salembier (1987a) propose to evaluate the needs which software design must meet, for each given work environment, on the basis of an understanding of the tasks and organization of the tasks which users have to perform. These authors distinguish two types of evaluation techniques used in software ergonomics: (1) "objective" or "direct" techniques by which an observer or device records events according to a predefined format ("necessary to obtain a reliable description of behaviours"); and (2) "subjective" or "indirect" techniques, where the operator is asked to describe the stages in performance of his task (through questionnaires, routing slips, conversations, etc.). The main goal of these methods is to produce detailed descriptions of the task of users or future users, either with a view to facilitating design of adapted software, or to identify corrections to be made to already installed technologies.

5.1.1 "Objective" or "direct" methods

"Objective" or "direct" methods are grouped in three categories. First, observation of the user's behaviour, the stages in performance of his task, where the observer has to carry out (consciously or unconsciously) a screening of the information available. (p. 22) The second category is assisted observation. This includes the observation grid method, whereby statistical operations are conducted on the data (frequency of use, distribution of work time, transition between types of tasks, time-based dynamics, etc.) and automatic recording methods (audio, video or "event sensors" at the peripheral level). Finally, one may use "trace studies", focusing on the result of the subject's activity (rather than the activity itself), in order to obtain indications on the type of activity. This is a complementary technique which, associated with subjective techniques, represents a methodological framework used, in particular, for the error analysis.

5.1.2 "Subjective" methods

According to Mazoyer and Salembier, the main usefulness of the questionnaire is for arranging a certain number of data in a hierarchical order, in view of a more thorough analysis, as well as providing elements useful for classifying tasks or workstations. The two authors do, however, mention two reservations concerning its use: the methodological rigour it requires with respect to sampling and distribution procedures; and the nature of the data gathered: these are, in effect, opinions and attitudes vis-a-vis objectives (technical, social, etc.), and as such they do not provide access to real behaviours[1].

Evaluation grids are a second subjective measure. These involve a type of questionnaire whereby users themselves evaluate a tool which they are using. The goal is to acquire a clearer view of a tool's strengths and weaknesses, through information provided by a specific category of users on the use of this tool. This method covers the

---

1. This type of reflection, wich recurs frequently in Mazoyer and Salembier, poses the problem of defining validity criteria. According to the authors, the observer is the only possible person responsible for defining what constitutes "real behaviour", his point of view being, conditionally with respect to a certain methological rigour, infallibly "objective". In fact, the point of view of the players ("subjects") may be based on constructions coloured by representations, their personal opinions concerning the task, their work environments, etc. On the other hand, a number of writers (in particular Guba and Lincoln, 1983, 1985) might say that the observer's analysis is also a construction based upon his representation of the user's task (with wich he is not necessarily very familiar) and his own task and objectives, and that it may be subject to bias owing to the relatively artificial context created by the observation situation itself. They would add that, consequently, his analysis must also be subject to a rigorous validation, that his findings, to be truly comprehensive, must be subject to a compromise between the observer's point of view and that of the subjects observed.

following aspects: functionality (most commonly used functions, least frequently used functions, identification of those lacking for execution of a specific task) and ease of use (vocabulary, command syntax, screen structure, protection against incorrect commands, documentation). Conversations and verbalizations generated apply through a more or less directive approach. In a non-work situation, the subject describes to the user what he does, the different stages in performing his task. Closer to observation methods, one may also use simultaneous conversations and verbalizations, where the subject explains to the observer what he does as he performs his task.

The last subjective method mentioned by the authors is that of "critical incidents", which, they say, is particularly well adapted to identifying dysfunctions, and may be seen as a specific, directed condition of the conversation. (p. 25) A sample of users are asked to report a work incident of their choice and evaluate it according to predetermined criteria (frequency, seriousness, impact on activity, etc.). The analysis then groups types of incidents together in a hierarchical order, and statistically processes the collected data. The authors make a parallel between this method and the "trace study" (see 5.1.1).

5.1.3 Other methods proposed

Another approach, known as self-confrontation, is a combination of direct and indirect methods. It involves asking the person to comment upon a video recording of his activity. Other techniques may also prove useful. The log is an example of this. It may be designed in two ways: (1) review of the official record, where in particular incidents and installation problems are noted as they occur; and (2) the user is asked, over a given work period, to record a certain number of facts and elements that may be predetermined (type of task at a given moment, site of activity, type of incidents noted, etc.). He may also be asked to report events by focusing on his own activity (what he did, how and why he did it, etc.).

A final method proposed by Mazoyer and Salembier is that of "information on demand": this involves placing a subject in a problem-solving situation, but providing him with no information at the outset; it is he himself who asks the experimenter for the information he considers necessary to perform the task. Requests for information enable the experimenter to rapidly identify the types of requests, and their order. (p. 25) The authors add, however, that an important limitation to this method lies in the tendency of subjects, when faced with the role of someone being observed, to formulate requests that are not representative of their usual activity and operating style.

5.2 Controlled environment evaluations and field studies

In reading the literature addressing the different aspects of the tool-user interface, one notes frequent references to two categories of evaluation methods, often presented as complementary: the **controlled environment evaluation**, and the **field study**. The idea is to combine different evaluation tools, in order to test the suitability and predict the impact of technologies in the pre-installation phase, or in order to evaluate adjustments that should be made to technologies that are already operational or in the process of being implemented. Generally speaking, controlled environment evaluation involves a laboratory study of the characteristics of the interaction between a predefined user group and the prototype of a tool or interface. The field study is more closely related to the needs evaluation proposed by Mazoyer and Salembier (1987a)[2]. This essentially involves evaluating tool-user interaction in an operational context, either through observation by the investigator, or based on the point of view of users, or a combination of the two methods.

In order to develop evaluation criteria that may be easily applied and generalized to installation contexts other than that on the basis of which they were implemented, Vanneste (1987) proposes developing a methodology based on a "modelling-prototyping" approach. He favours using such a method in the actual design of the tool-user interface

2. See 5.1 of this paper.

based on the following argument. The design work for these interfaces is confronted with the appearance of new needs: integrating applications at a single workstation, need to adapt interfaces to different classes of users (novice, expert, etc.), frequency of task interruption (moving to and fro between the main task and the different subtasks). Designing the human-computer interface thus becomes more complex, whence the interest in developing a modelling-prototyping test method.

The main goal of this methodology is to develop and validate the specifications of human-computer dialogue interfaces, taken independently of the "rest" of the computer tool to facilitate their adjustment and flexibility[3]. The author thus proposes isolating the characteristics specific to the interface, that is, the interaction styles: classical text mode (such as keyboard commands and alphanumeric display) and graphic mode (such as direct manipulation and multi-windowing), dialogue styles (question-answer, function keys, command language, menus, etc.), input-output techniques (display modes, etc.). The experimental approach and development of the model reflect poles of reference such as the characteristics of the tasks and users, different stages in the dialogue, command syntax, and the different system-user interaction devices.

---

3. We saw above (2.1) that there is controversy surrounding the evaluation approach that isolates the interface from its ramifications from the system.

According to Vanneste, all the elements necessary for testing dialogues (resources, dialogues and test runs) may be developed from the "development" functionalities of the dialogues (ie., the different dimensions and processes making up the dialogue). The framework of experiments and processing of results are defined from two poles. First, there are the "testing aids", which essentially concern definition and identification of the conditions of experimentation (type of models, type of users concerned, general context, etc.) and the configuration of the hardware and software, the assignment of a "test run" and designation of "trace factors"[4]. Follow-up on the experiment is provided by a log and by processing of the data gathered. Second, the modelling-prototyping outputs provide the data necessary for producing the specifications to be incorporated in the tool-user interface design.

Eason (1984) considers usability as a decisive criterion for evaluating acceptance of a computer tool by its users. In order to define this criterion in an operational context, he sets out the need for an evaluation process which gives the initiative to the user, in particular through observation of his responses to the use options offered to him by the tool (based on the rate of use of the different functions and possible operating procedures, etc.), his ways of building a learning (or non-learning, avoidance) strategy for these options. Such an evaluation procedure involves minimal control from the experimenter and an approach based on essentially qualitative data. The goal of this

---

4. See Mazoyer and Salembier, 1987a, pp. 22-23, and point 5.1.1 of this paper.

"subjective" approach is to highlight information concerning non-use of the tool rather than only its use.

One can be sure of the satisfactory usability of software and related documentation only when it is certain that the target user population will be able to learn to use each product with minimum difficulty and maximum efficiency (Neal and Simons, 1984). Evaluation of usability, Eason tells us, must take place in reference to the context[5] in which the tool is used, to avoid systematic generalization of results. One must therefore conduct "micro" analyses, spread over time, in order to clearly grasp users' changing needs and expectations as their expertise develops. This means not only a series of tests intended to correct problems, but also the verification of these corrections and their impact on the interface as a whole. The earlier these tests begin in the development of the tool, the more effective they will be and the easier it will be to make the necessary modifications.

In a similar vein, several authors suggest that evaluation of product usability be incorporated into the software design process and that it begin as soon as the developers' goals are defined (Neal and Simons, 1984; Mazoyer and Salembier, 1987b). It would thus cover all stages in specification, design, development, installation and use of the product, and continue throughout its useful life. Such an approach also has the advantage of providing a perspective for capturing the impact of

---

5. That is, the characteristics of the organization where the tool is installed, the types of tasks, the classes of users involved, their specific expectations, etc.

the context throughout the different stages of development of the technology, and thus of treating the tool-task-user-organizational dynamics as a whole.

In order to successfully cover all these stages in the design and use of the tool, Neal and Simons (1984), Mazoyer and Salembier (1987b) and Eason (1984) are in agreement in proposing a methodological approach integrating the field study, spread over a long period, and the controlled environment evaluation (or prototyping). The latter would help to compare the impact of different computer tools, standardize results and provide paths for addressing various contexts of evaluation.

Neal and Simons describe controlled environment evaluation as having to permit testing of usability according to various levels of complexity, in order to compare different design alternatives. To this end, the experimenter must ensure that various matches are established between the test situation and the actual work situation: (1) between the system supporting the prototype and the system that will actually be used to run the software; (2) between the sample of users operating the prototype and the class of users for which the interface is intended (with respect to the qualifications and technical knowledge required, motivation to use the tool, etc.); and (3) between the selected tasks and the actual tasks for which the tool will be used (corresponding in particular to the frequency of use of each function). The documentation (user manual, handbooks, tutorial software, etc.) concerning the product and the training for future users must be tested alongside the software itself. Thus the authors propose, among other things, recording the user's performances during and after training.

The goal of the field study, in Neal and Simons's perspective, is to complement the laboratory study since it identifies problems specific to integration of the tool into an actual work environment. To this end, they propose to measure the user's performances with a view to obtaining quantitative and qualitative information on three types of criteria: learnability, ease of use, and the difficulties encountered by the user. The authors propose grouping these criteria within what they call the playback methodology. This is an observation system which performs remote recording, timing and coding of the user's operations (in particular, errors, requests for assistance, and abnormally long times taken to execute an operation).

Unlike the observation or recording methods proposed by Mazoyer and Salembier (1987a)[6], the transparency of this tool minimizes the potential bias of direct observation of the user's activities in a work situation. In fact, this method, according to Neal and Simons, is meant to be "non-intrusive" to the extent that the data collection procedure does not infringe upon the user's normal activities due to the physical presence of an observer taking notes or through the inquisitive eye of a video camera. This method does, however, suppose that the evaluator is sufficiently familiar with the user's task to be able to place the data

6. See point 5.1 of this paper.

gathered in their context. Thus, an indicator such as frequency of data input is not necessarily a major element in the quality of the work.

5.3 Need for adaptability in the methods used

This rapid overview of the methodologies proposed by certain segments of the literature enable us to observe that evaluation of the user-friendliness of a computer tool may be built from several methods and several data collection sources. In this context, a procedure involving triangulation of several methodological tools, but also of data gathered at different moments in the design, development and installation of the tool[7], would make it possible to guarantee maximum soundness for the range of results obtained.

While we share the viewpoint of Mazoyer and Salembier (1987b) on the necessary complementary nature of the controlled environment evaluation and the field study, we must nonetheless stress the potential limitations of an approach based on essentially predetermined criteria, on too heavily standardized indicators. The reaction of a group of users to installation of a new tool depends upon numerous, varied factors. This complexity makes it difficult to provide a very precise definition of analytic criteria without a good prior general knowledge of the field characteristics, that is to say, the characteristics of the tool-task-user-organizational dynamics set. These are the characteristics, with

---

7. We refer here to the moments in the strategic decision to design the tool, the design, installation and development of the tool.

the needs, expectations, constraints and social relationships they translate, that determine which data-gathering instruments will be the most appropriate, whence the usefulness of the flexibility of the naturalistic approach.

In this perspective, the results of the evaluation may be generalized to other installation sites with respect to the conceptual approach used and the analytic methods and tools developed, but this is more difficult in relation to data gathered on site. Indeed, to produce applicable results, the evaluator's work involves not only indicating whether a computer tool is sufficiently user-friendly or not, but also establishing the factors and constraints which define the parameters of what is more or less user-friendly for users at an installation site. It is in light of such an overall view that evaluation may make it possible to determine the most satisfactory actions and tradeoffs for the organization as a whole.

CONCLUSION

The user-friendliness of the computer tool-user interface is thus a complex, relative concept which may be defined only in relation to the specific context in which the interaction occurs. Evaluation work must therefore involve understanding the elements and relationships forming the context of development, installation and operationalization of the tool. The latter is understood as being part of a **series** of elements which mutually interact as the installation process progresses: technology, tasks to be performed, users and organizational context are indissociably linked. This tool-task-user-organizational dynamics set, evolutionary in nature, and both social and technical, contains the basic ingredients of user-friendliness.

In this perspective, evaluation of user-friendliness takes place in an approach requiring a flexible methodological framework which can capture the social relationships, the players' representations and the characteristics of the work process as major factors in the suitability of the characteristics of the interface proposed by the tool. We suggest choosing a naturalistic-type evaluation approach, since this allows all these data to be gathered, through the contribution of a multiplicity of different methods and analytic tools. Such an approach ensures maximum soundness for the results obtained.

In practice, this means favouring a formative type evaluation approach. This is also an **inductive** approach: questions, specific goals and analytic criteria are constructed in interaction with the social players to hand, in line with their own expectations and apprehensions with respect to installation. The goal is first to **understand** the reasons underlying the effects generated by installation. Users are considered not as subjects, but as active **participants** in the evaluation, and in negotiating its approach. The evaluator adapts his approach, criteria and questions, to reflect the expectations and needs specifically expressed by users at the start of the process, when the evaluation plan is proposed. Players' opinions, "subjective" viewpoints, perceptions and representations are taken as essential information sources, since they contribute to constructing the context in which the tool will be used and an important portion of the factors on which action might be required to optimize the impact of installation.

Evaluation may achieve a rich, dynamic understanding of change by calling upon various sources of collection of quantitative and qualitative data. The conclusions drawn from raw data are submitted to the players for validation by the main parties involved and to obtain their point of view on how to interpret the findings of the investigation.

As we said at the outset, this short paper cannot lay claim to exhaustive coverage of the topic of the user-friendliness of interfaces, much less the issue of evaluation as a tool of technological and socio-organizational development. To validate and consolidate the proposed approach, it is necessary to ensure its extension in empirical testing, to test it in several case studies. We hope, however, that the reflections put forward here will provide avenues of research that may help us to better understand the social and human dimensions of technological development, and to develop intervention and management tools geared to the issue of introducing new work tools.

APPENDIX

ANNOTATED BIBLIOGRAPHY

BARNARD, P.J.; HAMMOND, N.V.; MORTON, J.; LONG, J.B.; CLARK, I.A. (1981);
"Consistency and compatibility in the human-computer dialogue", International Journal of Man-Machine Studies, No. 15, pp. 87-134.

It is to gain a better understanding of the relationships which a user may bring into play in dialogue with the system that the authors use their "block interaction model", which they present particularly as a means of classifying errors in system use. Based on a fusion of the analytic tools of human factors engineering with those of cognitive psychology, this model is used to specify four sources of potential influence on the user's representation of the system. First, the important issues in relation to the "internal" consistency of a set of commands is determined. Then the model shows to what extent general knowledge can be used to induce the principles which underlie command structure. Third, it aims to assess the issues in relation to compatibility of commands with "natural language". Finally, it must be used to identify the important issues concerning the user's knowledge of the machine, and to recognize what documentation and specific information displayed on a VDU should include. This model is illustrated by its application to three case studies on the human-computer dialogue.

BENNETT, John L. (1986);
"Tools for building advanced user interfaces", IBM Systems Journal, Vol. 25, No. 3/4, pp. 354-368.

Bennett describes user interface evaluation systems in terms of usability and meeting users' expectations. More specifically, he refers to development of User Interface Management Systems (UIMSs); the goal, operator, method, selection-rule (GOMS) model developed by Card, Moran and Newell; the Keystroke Model (a simplified version of the GOMS model) and the cognitive-complexity theory developed by Kieras and Polson. The author emphasizes the relevance of evaluation by **modelling**, based on an experimental system that may be more easily modified than a finished product and less costly than an evaluation after the system is built and put into use. The evaluation must be carried out on the basis of specific measurable and testable goals established by the builder and the evaluator representing the user. Bennett therefore encourages developers to separate design of the presentation and dialogue with the user from design of specific application content (separation of interface function from the rest of the system), at both the design and the evaluation stage.

CHRISTOL, J. (1987);
"Les logiciels, un travail pour l'ergonomie?" in ALERZA, C. *et al*, L'ergonomie des logiciels. Un atout pour la conception des systèmes informatiques. Les Cahiers technologie, emploi, travail, No. 4, October 1987, La Documentation Française, Paris, pp. 11-14.

Christol's argument is built around the observation that in evaluating the human cost of work and risks of error, the analytic tool and organization must reflect the existence of different users (novice mode and expert mode), in different states (security procedure more or less active, aids, division of tasks, etc.), faced with different situations (brief increase in workload, incidents, etc.). Evaluation of a technology therefore requires prior knowledge of the **context of operation**: workspace, physical and human environment, division of tasks between the user and the machine, division of tasks among employees and services, means of communication between users. In this perspective, the author proposes a list of means of minimizing the impact of the most frequent errors in computer systems use.

EASON, K.D. (1984);
"Toward the Experimental Study of Usability"; Behaviour and Information Technology, Vol. 3; No. 2; pp. 133-143.

Eason presents usability as a concept for measuring the extent to which a user can actualize the potential usefulness of a computer system. More concretely, it represents the gap between a system's potential use and the actual degree of use which unskilled users (ordinary people) are able or wish to achieve. Usability is described as the outcome of interaction among system, task and user. This interaction creates the context in which the user develops how he will set about carrying out a specific task.

The variables influencing a system's usability represent a series of implicit cost-benefit assessments conducted by the user. The author's aim is to examine the methodological considerations with respect to these variables, in order to contribute to making usability an analytic tool with application beyond experimental studies. The issue is to develop a model of the user's response to the requirements of the task, as this would indicate the extent of a system's usability. To this end, Eason proposes an approach whereby the initiative is given to the user, through minimal control by the experimenter and the use of qualitative data, in order to highlight information on what the system is not used for. Usability is defined in reference to the system's context of operation, whence the need for "micro" analyses spread over time. The author suggests using two types of complementary investigations: **field research**, spread over a long period; and **comparative studies** among different systems, conducted from simulations, which would contribute to standardizing results.

GIROUX, Luc; LAROCHELLE, Serge (1987);
L'ergonomie cognitive des systèmes informatiques : état de la question et pistes de recherche, University of Montreal and Canadian Workplace Automation Research Centre, July 1987, 154 p.

This study provides an overview of trends and the state of research in cognitive ergonomics of computer systems, leading to a proposal of avenues of research which the authors identify as the most likely to contribute to developing optimum interaction. The first two chapters present the overall issue of cognitive ergonomics, pinpointing it as a field of scientific investigation. Then the authors define what they identify as the major components of human-computer interaction: computer system, human user and task. Finally, they suggest **direct manipulation interfaces, study of users' mental models, development of intelligent interfaces and organizational interface** as avenues of research to be singled out for this field to advance.

GOODWIN, Nancy C. (1987);
"Functionality and Usability", Communications of the ACM, **March 1987, Vol. 30, No. 3, pp. 229-233.**

The author's goal is to demonstrate that functionality and usability are complementary system characteristics. Functionality represents the fit between a system's functions (or potential) and the tasks to be performed. Usability, in addition to being task-related, may be associated with users' characteristics (i.e., people-related), depending on their level of expertise, etc. It is thus defined in relation to a variety of characteristics, with respect to specific needs, the requirements of tasks and users, and how these needs evolve over time. Factors affecting usability may be of three types: organizational (design of training, accessibility of terminals, organizational culture, etc.), related to the features of the computer system (response times during periods of heavy usage) and associated with system design (potential and characteristics of its functions, user-machine dynamics, etc.). The author concludes by pointing out that designing a usable system requires understanding the intended users, their levels of expertise, the amount of time they expect to use the system, and how their needs will change as they gain experience.

LUPKER, Steven J.; FLEET, Gregory J.; SHELTON, Brian R. (1988);
"Callers' perceptions of post-dialling delays: the effects of a new signalling technology", Behaviour and Information Technology, **Vol. 7, No. 3, pp. 263-274.**

The authors' report on an experiment whose issue is to examine **"contrast effects"** with respect to the local telephone network which the reduction in post-dialling delays in the toll network may generate. In this perspective, their goal is to evaluate the negative impact of introducing a new technology giving rise to enhancement of the toll network on satisfaction with the conventional local network. Caller **impatience** and **abandonment** were thus evaluated under three systems: (1) the current system; (2) the current local system with a new toll system and (3) the current toll system with a new local system. The results of these observations, made from controlled environment simulations, will demonstrate that impatience and abandonment rates largely vary according to the impact of the "cognitive" contrast effect resulting from callers' greater expectations concerning system performance.

MAZOYER, B.; SALEMBIER, P. (1987a);
"Les principales techniques de recueil de données en ergonomie des logiciels", in ALERZA, C. *et al*, L'ergonomie des logiciels. Un atout pour la conception des systèmes informatiques. Les Cahiers technologie, emploi, travail, No. 4, October 1987, La Documentation Française, Paris, pp. 22-26.

The authors describe two types of data-gathering techniques geared to software ergonomics. First, they deal with "objective" or "direct" techniques whereby an observer or a device records events on the basis of a predefined analysis format. They then identify as "subjective" or "indirect" those techniques by which the operator is asked himself to describe the stages in performing his task (questionnaires, checklists, conversations). Mazoyer and Salembier also report on the "self-confrontation" approach (combination of direct and indirect methods involving asking the user to comment on a video recording of his activity), log techniques (where the user is asked, over a variable work period, to record a number of facts and elements) and "on-demand information" (where a subject is placed in a problem-solving situation, so that he asks the experimenter for the information he deems necessary to perform the task; this allows the experimenter to identify the types and order of importance of requests).

MAZOYER, B.; SALEMBIER, P. (1987b);
"La maniabilité : une dimension mesurable de la qualité des logiciels?", in ALERZA, C. *et al*, L'ergonomie des logiciels. Un atout pour la conception des systèmes informatiques. Les Cahiers technologie, emploi, travail, No. 4, October 1987, La Documentation Française, Paris, pp. 47-52.

The aim of this text is to review a usability evaluation methodology integrated into the software design and development process. The authors stress the importance of working with several complementary approaches and analytic techniques: **field** usability evaluation, with a view to demonstrating how users actually use computer systems; and **laboratory** usability evaluation, particularly through development of modelling tools, in order to construct realistic experimental situations where different product models or versions may be tested, specific tests carried out and changes implemented more quickly. Proposing an approach based on empirical criteria, in a wider perspective of software quality evaluation, the authors see problems in using the criteria generally proposed for assessing usability (such as homogeneity, flexibility, assistance to the user and learnability), primarily owing to the fuzziness of these concepts. These criteria are also criticized for involving "decision values" whose choice is left up to the evaluator. The authors also suggest that measurement of usability be limited to **output quality.** Indicators of specific problems could thus be highlighted, such as disproportionate execution times or inordinate error rates.

MORAN, Thomas P. (1981);
"An Applied Psychology of the User", ACM Computing Survey, March 1981, Vol. 13, No. 1, pp. 1-11.

Moran presents the psychology of the computer user as a subfield of computer science, approaching it from the perspective of the psychology of the user as individual (rather than social agent). He focuses on the cognitive aspects of the user, which include learning, performing, and reasoning (and not the motivational, emotional or personality aspects of the user). His approach, based essentially on empirical research, defines the user-friendliness of a system in terms of the user's activities, thus of the goals he has to accomplish, the constraints imposed upon him by the **structure of his task**, what he knows and his own mental information-processing limits. It is not a question of merely understanding user behaviour, but of successfully **predicting** and **controlling** it. To predict the user's behaviour, we must know its determinants for a given situation and calculate the optimal behaviour likely to satisfy them. Controlling user behaviour means manipulating these determinants by increasing the user's knowledge, providing him with technical or software support geared to his limitations, and organizing the task so as to better meet his goals. For Moran, there is no distinction between what are explicitly interface elements and the overall **internal structures** of the system prone to affecting the user's perception or conception of the system. The user thus develops a "conceptual view" of the system, from the total behaviour of the system. This **"user's conceptual model"** is, according to Moran, an integral part of the user interface. It is a product of the user's knowledge (or representation) of how the system works and how it can be used to accomplish his tasks.

NEAL, A.S.; SIMONS, R.M. (1984);
"Playback: A method for evaluating the usability of software and its documentation", <u>IBM Systems Journal</u>, Vol. 23, No. 1, pp. 82-96.

This paper describes a methodology for obtaining objective measures of system usability by collecting performance data on the user interface. The goals pursued in order to obtain satisfactory usability are achieved when it is certain that the target user population will be able to learn to use each tool with minimum difficulty and maximum efficiency. The user-friendliness (or ease of use) criteria vary with the class of users for which the product is intended. To evaluate usability, the authors propose a two-phase approach: (1) definition of the specific objectives and scope of the test with respect to the different aspects of the product, such as command syntax, task goals, menus, messages, on-line help facilities, training, system response time, etc; and (2) the usability test proper, which involves integrating two evaluation methodologies: laboratory evaluation of the interface (in a controlled environment, through modelling or prototyping); and field study of its application. To make the controlled environment evaluation as reliable as possible, Neal and Simons propose application of the playback methodology, based on a system which conducts remote recording, timing and coding of the operations carried out by a user sample, in order to identify such indicators as errors, requests for assistance and abnormally long times taken to execute an operation. This method is meant to be non-intrusive to the extent that data collection does not infringe upon the user's normal activities by keeping an operating log or due to the physical presence of an observer.

NORMAN, Donald A. (1983);
"Design Principles for Human-Computer Interfaces", <u>Proc. Chi'83, Human Factors in Computing Systems</u>, Boston, 1983, ACM, New York, pp. 1-10.

The author approaches the human-computer interface from the perspective of human factors in computer systems, which he presents as a new field of scientific practice. The criteria by which preferred design principles should be articulated are **generality** (possibility of applying these principles to more than a single context) and **precision** (so that they are applicable to a real context). Norman favours taking into consideration the characteristics (powers and weaknesses) of the technology itself, tied to a close relationship with local constraints (technical, organizational, budget- or time-related, etc.). Design problems do not make for simple answers, but only **tradeoffs** between the different system constraints (exchanges between chosen powers and weaknesses), each of whose elements must be interpreted with reference to their specific context. Human factors analysis must make these tradeoffs explicit and demonstrate how they interact with the technology itself in the design process. These tradeoffs are defined in terms of the choices offered by the technologies used, the classes of users, design objectives and gains and losses of the interface with respect to the choices to be made. Their analysis is presented as a quantitative design tool for a system-user interface which can be used as a frame of reference in designers' choice. The author proposes separating the interface design from other programming tasks (for instance, by forming a separate data module), with a view to allowing greater flexibility, both to the designer and to the interface itself, and offering the possibility of creating personalized interfaces.

PLANAS, Miguel A.; TREURNIET, William C. (1988);
"The effects of feedback during delays in simulated teletext reception", <u>Behaviour and Information Technology</u>, Vol. 7, No. 2, pp. 183-191.

The objective here is to examine the possibilities of explaining the annoyance expressed by users of a teletext service and evaluating the variability of the reported annoyance (impatience) rates, depending on the type of feedback offered by the system and the duration of the operation. It presents the design, methodology and results of two laboratory (controlled environment) experiments whose goal was to study users' perception of different types of feedback. In conceptual terms, two categories of feedback are identified: **system feedback** (when the system informs the user that a search for the requested page is in progress); and **time feedback** (where a message indicates how long the user must wait to obtain the information requested).

SHARIT, Joseph; CUOMO, Donna L. (1988);
"A cognitively based methodology for evaluating human performance in the computer-aided design task domain", <u>Behaviour and Information Technology</u>, Vol. 7, No. 4, pp. 373-397.

The authors' objective is to develop a methodology for evaluating human performance in the computer-aided design (CAD) task environment. The basic question is: how does technological development in this CAD sector translate into user performance with respect to (1) different types of tasks; and (2) different levels of complexity for a particular task. The aim is therefore to develop tools to evaluate compatibility between human skills and CAD systems. The approach is based on five long-term research goals: (1) developing methods for evaluating mental workloads and methods for reducing these workloads; (2) understanding better the cognitive capabilities of the human in relation to the cognitive demands of the tasks and the interfacing tools used to perform the tasks; (3) developing data summarization techniques which could serve to accomplish the above objectives; (4) developing performance evaluation methodologies, including **error analysis** (the system-user interface is represented by the input of a language structure which the authors break down into four levels: conceptual, semantic, syntactic and lexical. Errors, depending upon the input level they occur in, have different impacts and meanings); and (5) creating a classification system for computer tasks which would facilitate evaluation of human cognitive capabilities and limitations. The authors approach the human-computer link through the observation of fundamental interaction tasks (FITs).

SPÉRANDIO, J. (1987);
"Introduction à l'ergonomie des logiciels", in ALERZA, C. *et al*, L'ergonomie des logiciels. Un atout pour la conception des systèmes informatiques. Les Cahiers technologie, emploi, travail, No. 4, October 1987, La Documentation Française, Paris, pp. 15-21.

Spérandio presents here the main aspects of software ergonomics. Overall, it defines the **functionality** that the software must provide, depending on the type of task and type of users, establishes its fit with users' mental representations, selects **conditions for** computer-user **dialogue**, and optimizes information coding, for both presentations (visual in particular) and commands. According to the author, the variety of **users, tasks** and **needs** (motivations) with respect to a single type of technology points to the need for a work analysis prior to software design. The role of such an analysis is to define the most suitable functionality, coding and dialogue methods, in line with distinctions between novice, occasional or experienced users and between infrequent, frequent or repetitive tasks.

STEVENS, G.C., (1983);
"User-friendly computer systems? A critical examination of the concept", Behaviour and Information Technology, Vol. 2, No. 1, pp. 3-16.

This paper is a critical analysis of the concepts underlying the term "user-friendly", in order to show that, as at present defined, it is not helpful to system designers. For Stevens, the user-friendliness of a system is not a one-dimensional concept inasmuch as it may be defined in relation to **different classes of tasks and users**, and therefore must cater to diversified needs, expectations and satisfaction criteria. Prominence is given to changes to be made in approaches to interface design in order to clarify and broaden the criteria for defining the concept of user-friendliness. Against this background, the author offers a review of the main elements used to define user-friendliness: ease of use, naturalness of the interface, ease of understanding and helpfulness for the user. The problem Stevens notes for each of these concepts is their inability to grasp the variability and evolutionary nature of users' skills and needs. In conclusion, the author points to the need to pay more attention: (1) to variation in skill among users and, over time, in the same user; (2) to using expert users as intermediaries between the information user and the system; and (3) the nature and achievement of user motivation.

STORRS, G. (1989);
"Towards a Theory of HCI. A conceptual model of human-computer interaction?", Behaviour and Information Technology, Vol. 8, No. 5, pp. 323-334.

Graham Storrs's objective is to present the basis of a theoretical model of human-computer interaction (HCI) to assist practitioners in developing an overall understanding of the topic. The author defines the components of his model. Participants in the interaction are described as **autonomous agents** (ie., their motivations are not determined by other agents), **rational** (they are able to reason) and **sensible** (through their perception of the external world). Interaction with the computer (described as an artefact) is conditioned by the **purposes** of the agent (or group of agents) using it. In this connection, the distinction is made between interaction and interface. An interaction is an exchange of information by which two agents each modify the state of the other (i.e., the composite of the states of an agent's beliefs, goals, plans, intentions, attitudes, values, dispositions and physical state), while an interface is the set of information channels which support and influence (through their characteristics and the constraints they present) the content and structure of the exchange. In this perspective, the user interface is not something which can be treated separately from the rest of the system, inasmuch as there is no clearly defensible place to draw the line between front-end and back-end systems. For the author, the answer to the need for intelligent interaction lies not in the interface itself, but in using a non-human agent which acts as an intermediary to an interaction between two other agents. Broadly speaking, such **intermediaries** are messages, codes and options, adaptable structures enabling user-agents to make their way within the interface, to better understand its constraints and potential, in order to translate their aspirations and "states" more satisfactorily.

TYLER, S.; TREU, S., (1989);
"An interface architecture to provide adaptive task-specific context for the user", International Journal of Man-Machine Studies, No. 30, pp. 303-327.

Using the concepts of learnability and usability, Tyler and Treu point out that interface design faces requirements from users that are varied and change over time. In this perspective, they propose an "adaptive" interface design, based on the "modularity" of the interface system, which resides in a separate module that may be easily modified independently of the rest of the programming. The user is thus offered the opportunity to "assemble" and modify the content and presentation of the assistance with which the interface can provide him (error messages and operating instructions), in a personalized way, on the one hand depending on the "style" of the task and the level of knowledge of the system with which he is the most comfortable. The philosophy underlying this interactive interface design is to promote learnability rather than user-friendliness, to encourage non-expert users to learn and give them the opportunity to evolve in their use. Here, user-friendliness is seen in terms of adaptability and the need to respond to variable needs.

VANNESTE, C. (1987);
"Maquettage et prototypage pour la conception d'interfaces homme-ordinateur", in ALERZA, C. *et al*, L'ergonomie des logiciels. Un atout pour la conception des systèmes informatiques. Les Cahiers technologie, emploi, travail, No. 4, October 1987, La Documentation Française, Paris, pp. 53-56.

The author demonstrates that the design of human-computer interfaces is confronted with the appearance of new needs associated with integrating applications at a single workstation, adaptability of interfaces to different classes of users (novice, expert, etc.), and "interruptibility" of tasks (moving to and fro between the main task and the different subtasks). Designing the human-computer interface thus becomes more complex, whence the interest in using a **modelling-prototyping tool**. The main goal of such a tool would be to develop and validate the specifications of human-computer dialogue interfaces alone (i.e., to isolate the interface from the purely technical "internal" aspects to facilitate their adjustment and flexibility). The experimental approach and development of the model (prototype) are based on the characteristics of the tasks and users, the different stages in the dialogue, command syntax, and the different interaction devices of the human-computer interfaces.

BIBLIOGRAPHY

BARNARD, P. J.; HAMMOND, N. V.; MORTON, J.; LONG, J. B.; CLARK, I. A. (1981);
«Consistency and compatibility in the human-computer dialogue», International Journal of Man-Machine Studies, 1981, No.15, 87-134.

BENNETT, John L. (1986);
«Tools for building advanced user interfaces», IBM Systems Journal, Vol. 25, No. 3/4, 354-368.

CHRISTOL, J. (1987);
«Les logiciels, un travail pour l'ergonomie?» in ALERZA, C. *and al.*, L'ergonomie des logiciels. Un atout pour la conception des systèmes informatiques. Les Cahiers technologie, emploi, travail, No. 4, October 1987, La Documentation française, Paris, p.11-14

EASON, K. D. (1984);
«Toward The Experimental Study of Usability», Behaviour and Information Technology, Vol. 3, No. 2, p.133-143.

GIROUX, Luc; LAROCHELLE, Serge (1987);
L'ergonomie cognitive des systèmes informatiques: état de la question et pistes de recherche, Université de Montréal and Canadian workplace Automation Research Center, July 1987, 154 p.

GOODWIN, Nancy C. (1987);
«Fonctionality and Usability», Communications of the ACM, March 1987, Vol. 30, No. 3, p. 229-233.

GUBA, E. G. (1987);
«What Have we learned About Naturalistic Evaluation?», Evaluation Practice, Vol. 8, No.1, Feb. 1987, Newbury Park, p. 23-43.

GUBA, E. G.; LINCOLN, Y. S. (1981);
Effective Evaluation : Improving the Usefulness of Evaluation Results through Responsive and Naturalistic Approaches, Jossey-Bass Publishers, London, 423 p.

JODELET, D. (1988);
A propos des représentations, allocution in the «Séminaires du Centre de recherche en évaluation sociale des technologies», Montreal, 25 march 1988.

KLEIN, H. K.; HIRSCHHEIM, R. A. (1989);
«Legitimation in information systems development: a social change perspective»; Office Technology and People, Vol. 5, No. 1, p. 29-46.

KLING, R.; IACONO, S. (1989);
«The institutional character or computerized information systems», Office Technology and People, Vol. 5, No. 1, p. 7-28.

LINCOLN, Y. S.; GUBA, E. G. (1985);
Naturalistic Inquiry, Sage, Londres, 1985, 416 p.

LUPKER, Stephen J.; FLEET, Gregory J.; SHELTON, Brian R. (1988);
«Caller's perceptions of post-dialing delay : the effects of a new signalling technology», Behaviour and Information Technology, 1988, Vol. 7, No. 3, p. 263-274.

MAZOYER, B. ; SALEMBIER, P. (1987 a);
«Les principales techniques de recueil de données en ergonomie des logiciel» in. ALERZA, C. *and al.*, L'ergonomie des logiciels. Un atout pour la conception des systèmes informatiques. Les Cahiers technologie, emploi, travail, No. 4, October 1987, La Documentation française, Paris, p. 22-26.

MAZOYER, B. ; SALEMBIER, P. (1987 b);
«La maniabilité: une dimension mesurable de la qualité des logiciels?» in ALERZA, C. *et al.* L'ergonomie des logiciels. Un atout pour la conception des systèmes informatiques. Les Cahiers technologie, emploi, travail, No. 4, October 1987, La Documentation française, Paris, p.47-52.

MORAN, Thomas P. (1983);
«An Applied Psychology of the User», Computing Survey, March 1983, Vol. 13, No. 1, p. 1-11.

NEAL, A. S. ; SIMONS, R. M. (1984);
«Playback: a method for evaluating the usability of software and its documentation», IBM System Journal, Vol. 23, No. 1, p. 82-96.

NORMAN, Donald A. (1983);
"Design Principles for Human-Computer Interfaces", Proc. CHI'83. Human Factors in Computing Systems, Boston, 1983, ACM, New York, p. 1-10.

PLANAS, Miguel A.; TREURNIET, William C. (1988);
«The effect of feedback during delays in simulated teletext reception», Behaviour and Information Technology, Vol. 7, No. 2,
p. 183-191.

SHARIT, Joseph; CUOMO, Donna L. (1988);
«A cognitively based methodology for evaluating human performance in the computer-aided design task domain», Behaviour and Information Technology, Vol. 7, No. 4, p. 373-397.

SPÉRANDIO, J. C., (1987);
«Introduction à l'ergonomie des logiciels» in ALERZA, C. *et al.*, L'ergonomie des logiciels. Un atout pour la conception des systèmes informatiques. Les Cahiers technologie, emploi, travail, No. 4, October 1987, La Documentation française, Paris, p.15-21

STEVENS, G. C. (1983);
«User-friendly computer systems? A critical examination of the concept», Behaviour and Information Technology, Vol. 2, No. 1, p. 3-16.

STORRS, G. (1989);
«Toward a theory of HCI. A conceptual model of human-computer interaction», Behaviour and Information Technology, Vol. 8, No. 5, p. 323-334.

TYLER, S. ; TREU, S., (1989);
«An interface architecture to provide adaptative task-specific context for the user», International Journal of Man-Machine Studies, No. 30, p. 303-327.

VANNESTE, C. (1987);
«Maquettage et prototypage pour la conception d'interfaces homme-ordinateur», in ALERZA, C. *et al.*, L'ergonomie des logiciels. Un atout pour la conception des systèmes informatiques. Les Cahiers technologie, emploi, travail, No. 4, October 1987, La Documentation française, Paris, p. 53-56.

YOUNG, R. M. (1981);
«The machine inside the machine: user's models of pocket calculators», International Journal of Man-Machine Studies, No. 15, p. 51-85.

Pour plus de détails,
veuillez communiquer avec :

Le Centre canadien de recherche
sur l'informatisation du travail
1575, boulevard Chomedey
Laval (Québec)
H7V 2X2
(514) 682-3400

For more information,
please contact:

Canadian Workplace
Automation Research Centre
1575 Chomedey Blvd.
Laval, Quebec
H7V 2X2
(514) 682-3400