International Review of Industrial and Organizational Psychology 1987
Edited by C. L. Cooper and I. T. Robertson

Chapter 5

HUMAN-COMPUTER INTERACTION IN THE OFFICE

Michael Frese
Department of Psychology
University of Munich
West Germany

INTRODUCTION

There is no doubt that computers have already changed work and will continue to do so in the future. According to some estimates, 40–50 per cent of all American workers will deal daily with a computer by the end of this century (Giuliano, 1982). The following technological trends are likely to appear in the future (see, for example, Otway and Peltu, 1983):

— stronger interconnectedness of computers;
— a proliferation of software so that many different programs are at our disposal and many will have to be learned;
— an increasing integration of software so that a writer can interchange between, for example, graphics, (business-)calculations, using a database, putting a thought on a notepad, and writing a text;
— an integration of traditional uses of the computer and the telephone (teleconferencing, electronic mail and mailbox, voice-mail, telecopying, etc.);
— an increasing use of decision support systems (e.g. in the work of the insurance sales person);
— the use of huge databases that are integrated into daily work; additionally, filing will be done electronically.

In spite of the obvious importance of this new technology, there is curiously little interest among industrial and organizational psychologists in dealing with the topic of human-computer interaction at the workplace; there are very few contributions on this topic in the major journals (e.g. *Journal of Applied Psychology*). On the other hand, there is actually ample literature on psychological issues in human-computer interaction as documented in two literature guides (Rödiger, 1985; Williams and Burch, 1985) but this research was mainly done within the frameworks of human factors and cognitive science (as in the three journals *International Journal of Man-Machine Studies, Behavior and Information Technology,* and *Human-Computer Interaction*). In addition, there has been a score of edited volumes on this topic, e.g. Balzert, 1983; Bennett, Case, Saudelin, and Smith, 1984; Borman and Curtis, 1985;

Bullinger, 1985; Cakir, 1983; Green, Payne, and Veer, 1983; Janda, 1983; Mantei and Orbeton, 1986; Norman and Draper, 1986; Schauer and Tauber, 1984; Shackel, 1985a; Thomas and Schneider, 1984; Veer, Tauber, Green, and Gorny, 1984; also, several textbooks have appeared, e.g. Monk, 1984; Nickerson, 1986; Shneiderman, 1980).

The emphasis here will be on computer-related work in the office (of course, human-computer issues cannot be divided neatly into those that relate to the office and those that do not). I have concentrated on European and American research that was published after 1980. Industrial and organizational psychology is usually interested in the study of organizational problems, long- and short-term difficulties for the workers (stress), how one can improve productivity, individual differences, and in recommendations for job design. Therefore, these are the topics of this review: What are the organizational conditions of human-computer interaction? What are potential stress problems? How can the human-computer interaction be optimized (in terms of improving the system and training the worker)? The question of individual differences: how can industrial psychologists influence the design process and what kind of guidelines are useful? Additionally, some general ('big') controversies that pervade different issues of human-computer interaction will be elaborated.

Actually, the emphasis on office automation would also call for reviewing the literature on combining microprocessors with the telephone (telecommunication), but this could not have been done within the space constraints of this chapter. This is a pity because telecommunication will probably change society and the way we live and work more than the computer alone. The following topics were also *not* pursued although they may be touched upon at times: hardware ergonomics (cf. Cakir, Hart, and Stewart, 1979; Monk, 1984; Nickerson, 1986; Spinas, Troy, and Ulich, 1983), robotics, CAD/CAM systems (computer-aided design, computer-aided manufacturing) or computer-integrated manufacturing, management information systems, and psychological issues in programming (Schauer and Tauber, 1983; Weinberg, 1971). Expert systems (Hayes-Roth, Waterman, and Lenat, 1983) were also not considered except as a tool in supporting human-computer interaction.

THE 'BIG' CONTROVERSIES IN HUMAN-COMPUTER INTERACTION

There are essentially three 'big', partly overlapping controversies: (1) low-level vs. high-level approaches; (2) human control vs. machine control at work; and (3) computer as a normal (albeit complicated) tool vs. as something new. I call these controversies 'big' because they pervade several different specific areas of research and application and because they stem from deep philosophical and value differences.

Level-of-Analysis Controversy

The level-of-analysis-controversy is curiously reminiscent of the debate on a molecular vs. molar learning theory between Hull and Tolman about 50 years ago. The molecular school of thought argues for an analysis of the lowest level, combining this with precise operational measurement, and mathematical laws. For example, Card, Moran, and Newell (1983) and Newell and Card (1985) argue for this kind of low-level analysis that 'there is a small number of information-processing operators, that the user's behavior is describable as a sequence of these, and that the time the user requires to act is the sum of the times of the individual operators' (Card *et al.*, 1983, p. 139). They describe the human as consisting of long-term memory, working memory (i.e. visual image store and auditory image store), the perceptual processor, the cognitive processor, and the motor processor. Each processor needs a certain amount of time (specified in milliseconds); e.g. the cognitive processor needs about 70 msec. So, for example, the time a person needs for pressing 'yes' when two symbols are identical and 'no' when they are not is calculated to be 310 msec after the presentation of the second symbol: 1 perceptual processor for perceiving the second symbol (100 msec) + 1 cognitive processor for matching first and second symbol (70 msec) + 1 cognitive processor for deciding what to answer (70 msec) + 1 motor processor (70 msec). Their keystroke model is similarly elementaristic; it proposes, for example, that in order to evaluate a move on a word processor, the number of keystrokes should be counted (and when designing it, minimized). Card *et al.*'s argument for their approach seems to be threefold: (1) an approximate quantification is better than none; (2) only a hard science approach (providing quantified laws) will be accepted by designers; and (3) this approach allows for an analysis of design alternatives *before* they are actually designed.

An alternative school of thought emphasizes molar, high-level approaches (Norman, 1986b; Carroll, 1986; Greif, 1986a). This is well exemplified in the volume edited by Norman and Draper (1986). Although the editors pursue a pluralistic approach, the contributions are usually high-level ideas, metaphors (e.g. design in architecture and design of interfaces), and paradigms rather than detailed, ready-to-use, quantifiable, low-level concepts. An example is Norman's (1986a) chapter on cognitive engineering. He specifies an action theory framework consisting of goals, intentions, action specifications, execution, perception, interpretation, and evaluation. These are high-level concepts that cannot be measured in milliseconds. He then goes on to contrast the designer's model with the user's model, emphasizing the differences and the fact that the user cannot directly recognize the designer's model, but can only work via the system's image. Clearly, this approach lacks the detail of Card *et al.*'s but may be more applicable to workplace issues.

Human Control vs. Machine Control

The second 'big' controversy—human control vs. machine control—is not yet out in the open but looms behind many issues (Boddy and Buchanan, 1982; Brown, 1986). It is related to the question of which kind of division of functions between human and machine should be aimed for (Price, 1985). Personal control can be defined as having an impact on the conditions and on one's activities in correspondence with some higher-order goal. (Frese, 1984b, in press, a). This implies that people are able to decide on their goal, their plans to reach the goal, the use of feedback, and the conditions under which they work. These decisions may refer to the sequence of how one does things, the timeframe (how quickly and when) and the content of the goals, plans, use of feedback, and conditions (Frese, in press, a). There are several factors which can be conceptualized to be prerequisites of a sense of control although they are not identical to control: (1) transparency of the system

(Maass, 1983; Brown, 1986) (however, a system may be transparent but not controllable), (2) predictability of the system (however, it may be predictable without being controllable), (3) functionality of the system, i.e. it is possible to achieve one's goals with the tools at hand (however, functionality does not assure controllability), and (4) skills concerned with how to develop plans and put them into action (however, one may be skillful but the system still does not allow decision points).

A very clear statement on the controversy between human control and machine control for the blue-collar sector has been made by Kern and Schumann (1984). In their sociological study of managing new technology in the machine-producing sector, they distinguished two production 'philosophies': a non-ideological-empirical approach to rationalization and a narrow-minded, technological approach. The latter implies that the technology is driven to its limits so as to automatize as completely as possible. The rest that cannot (yet) be done by technical means or is uneconomical to do with machines is left to humans (e.g. putting the raw material into the computer-driven lathe). The former approach takes into consideration the knowledge of the skilled workers and tries to teach the necessary skills to master the computer-driven lathe, e.g. teaching programming or working together with the programmers on the problems at hand. There is evidence that the non-ideological position is more functional, at least for reaching production goals in Western societies (cf. also Corbett, 1985).

One area in which there is a particularly lively debate on the issue of control is piloting airplanes. There is a call for actually reducing the achieved level of automation because this level might have led to a 'sterile cockpit' with ensuing accidents (Sundermeyer and Haack-Vörsmann, 1983; Wiener, 1985; Wohl, 1982). This is reminiscent of the old concept in industrial psychology where the human being should be an 'active operator' *vis-à-vis* automated systems (Hacker, 1978).

A similar reasoning applies even to expert systems (e.g. Coombs and Alty, 1984). Traditional conceptualizations of expert systems (e.g. Feigenbaum and McCorduck, 1984) have emphasized that expert systems are better than experts and should replace them or at least tell them what to do next. In contrast to this is the position that expert systems should give advice to the expert (e.g. via large databases), but that the expert should always be in control. One reason for the latter position lies in the fact that experts at the workplace typically have legal and moral responsibilities; only human beings can take responsibility (Fitter and Sime, 1980) and only they will care in a moral sense about what happens as a result of their expertise (Sabini and Silver, 1985). While the standard argument is that the responsibility for an expert system may lie with the programmer, complicated systems are usually not transparent even for the programmer. Fitter and Sime quote cases in which automation engineers who had designed the system had to experiment with it to find out how the system worked. When expert systems become truly able to learn, the issue of responsibility will become hopelessly muddled.

The issue of control comes up in the use of computers in the office as well (Kaye and Sutton, 1985). Gregory and Nussbaum (1982) maintain in their pessimistic review that the increasing introduction of computers will lead to more machine control, tighter (machine-) supervision, social isolation and little freedom of movement, as well as the deskilling of machine operators. To support their position they quote published IBM projections and Glenn and Feldberg's (1977) paper. A good example of attempting to use the computer to streamline social contact in the office, and thus reduce control is the suggestion by Cashman (1985) that the 'coordinator tool' should allow social contact only when this is in line with the official tasks that a person is supposed to do. Bjorn-Andersen (1983) found that there was a higher degree of structure, preprogramming, and formalization in computerized banks leading to a reduction in choices for the individual worker. Buchanan and Boddy (1982) found in their case study of word processing that management control became tighter after the introduction of computers. However, all of them agree that this is a management decision and not a necessary consequence of using computers (also Ellis, 1984; Schardt and Knepel, 1981; Spinas, in press). Similarly, Cornelius (1985) maintains that it is an organizational choice whether there is more machine or human control and he paints several scenarios for a bank office in 1990 which differ from one another in control and skill utilization. Pava (1983) concurs with this line of argument and describes how office work design is usually organized according to two rationality criteria: (1) streamlining office work in some linear fashion and (2) optimizing discrete components of office work (e.g. typing and telephoning) by introducing new machines and/or new organizational approaches. He points out several problems with these approaches and advocates a sociotechnical design concept for the office in which control is enhanced.

Thus, the issue of control comes up again and again in organizational design of computer work, as well as in the use of computers as tools and in usability guidelines (cf. Benbasat and Wand, 1984; DIN 66234, 1984; Frese, in press, a). Controllability is important because it has an impact on the functionality, the usability, and the user friendliness of the system. The question of control may, however, also be dependent on the user. Novices (or infrequent users) of a specific system may at first need some guidance from the system, while experts want control over the system and can make use of it (Benbasat and Wand, 1984; Shneiderman, 1980).

Computers as Tools?

The last 'big' controversy—the computer as a normal tool or as something new that has a larger impact on our being, thinking, and feeling than earlier technologies—has been debated philosophically, starting with Turing's (1950) test of artificial intelligence and leading to Turkle's (1984) discussion of the computer being like a 'second self'. Human factor workers side more with conceptualizing computers as just another (albeit quite complicated and powerful) tool, while scientists in the areas of artificial intelligence and some social critics are in the other camp. Weizenbaum (1976) warned that increased computer use will lead to a reduction in qualititive reasoning because it will be replaced by thinking mainly about problems that can be easily quantified. Taking this critique one step further, Volpert (1985) argues that the employment of computers and particularly expert systems at the workplace will lead to intellectual and creative deficits, to a Taylorization of expert work, and to a reduction of social relationships; and that computers offer an easy dream world to those individuals who have problems finding social and responsible contacts with others.

This view is to be contrasted with the concept that computers are tools that can have positive or negative consequences depending upon the organizational conditions

under which they are used and the sophistication of hardware and software design (e.g. Dzida, Hoffmann, and Valder, 1984; Ulich and Troy, 1986). This position is implied in the work of most human factor researchers and industrial and organizational psychologists since recommendations, guidelines, better design, improved functionality and usability, and finally, higher user friendliness should decrease negative effects and make computers more usable in working on one's tasks.

ORGANIZATIONAL FRAMEWORK AND CONSEQUENCES OF COMPUTER USE

Acceptance of Computer Use in the Organization

In many European countries, particularly in West Germany, computers were looked upon with some suspicion by the general public and by the blue- and white-collar workers (Lange, 1984; v. Rosenstiel, 1984). Therefore, attitudes to and acceptance of computer technology received some attention (cf. Helmreich, 1985; Reichwald, undated). The following factors seem to increase acceptance:

- prior knowledge of computers (Frese, 1984a; Hiltz, 1983);
- participation when introducing new technology at the workplace (see also next chapter);
- good training (this is so important that we deal with it in a separate section);
- good hardware ergonomics (Radl, undated);
- concrete working conditions after introduction of computers, particularly control at work (Eller, 1984), job complexity (Müller-Böling, 1984), and low stress levels (e.g. small response time, machine does not break down often; Müller-Böling, 1984);
- no loss of qualifications with introduction of computers (Weltz, undated);
- anticipation of positive personal and social consequences of computer use, e.g. better working conditions, personal advancement (Frese, 1984a).

Introducing Computers at the Workplace

There is no doubt that the process by which management introduces the computer at the workplace is crucial to whether the computer system will be accepted or whether there will be resistance to change. Resistance to change seldom manifests itself in open revolt (e.g. strikes, sabotage). More often there are indirect measures of resistance, e.g. reduction of output, keeping information to oneself, working strictly according to work rules, and not using the new system (Baroudi, Olson, and Ives, 1986; Hirschheim, Land, and Smithson, 1985; Weltz, undated). The psychological theory of reactance (Wicklund, 1974) may best explain these phenomena. Reactance appears when freedom is taken away and control is reduced. A further consequence of non-control is learned helplessness (Seligman, 1975). Wortman and Brehm (1975) have argued that the first response to non-control is reactance, and that after a period of time in non-control situations helplessness ensues. Since helplessness has cognitive, motivational, and behavioral consequences, a helpless worker may not be able to learn to use a new system, or may not be motivated to get to know it, and may stay passive—not in the sense of not working, but in the sense of trying to avoid the use of new technology.

Ever since the publication of the classic paper by Coch and French (1948), the standard answer of organizational psychologists to reduce the problem of reactance and helplessness is participation in the process of introducing new technology. Even though there is near unanimity among social scientists on the importance of participation, this recommendation is often not heeded in practice (Gottschall, Mickler, and Neubert, 1985; Spinas, in press; Sydow, 1984). Unfortunately, when participation is tried, it is often employed too late and there is too little of it (Weltz and Lullies, 1983). It is too late because the reactance frequently starts when the very first steps are made by management: e.g. analysis of current situation or needs assessment. At this point rumors and anxieties arise, as well as feelings that things cannot be influenced any more because everything has already been decided (Weltz and Lullies, 1983). Anxiety is usually high when workplace change is anticipated since old routines and old skills may become obsolete, the social situation may change and consequently a loss of social status is feared, and finally, there is always the threat of unemployment if this is not clearly and convincingly denied by management (Frese and Wendel, in preparation). Older people, particularly, may be fearful about whether they are still able to learn a completely new system. Thus, participation has to start right after management begins to deliberate on technological changes, even if it is not known at this point what the changes will look like. User participation takes time and training and often the system is ordered before the users can effectively participate (Eason, 1982).

There is too little participation because the professional organizers of technological change try to make a perfect plan that allows little end-user influence (Weltz, undated). They do this because they are under pressure to produce 'cost-effective results'. These savings are often compared to unrealistic estimates by the companies selling hardware and software. The fact that there is too little and too late preparation leads to passive resistance. This in turn can force management to use more authoritative measures: a vicious circle results.

Another problem is that end-users do not know enough about the technology to be able to effectively participate. Weltz (undated) argues that there are two types of experts, the system's expert and the 'usage' expert. The 'usage' expert's interest in the system concerns only its tool aspect. The end-users are usually the ones who have higher expertise in the usage aspect of the system because they know the tasks and how to do them. Unfortunately, however, the system's experts tend to dominate the introductory process which may lead to non-optimal solutions from a usage point of view. Thus, participation of end-users can also enhance the system's functionality.

Therefore, the following steps in participation should be followed (see Briefs, Ciborra, and Schneider, 1983, and Mumford, 1980, for case studies and Spinas, Troy, and Ulich, 1983, for guidelines):

- People should be trained for participation.
- Participation should start at the point of needs assessment and feasibility study (Johansen and Baker, 1984), e.g. with a subjective activity analysis (Ulich, 1981).
- A system should be installed on a trial basis so that it can be elaborated and amended (Eason, 1982).

- Early information should be given; it should be understandable, i.e. it should be concrete, related to the concrete tasks and work procedures—and not technically oriented (Weltz, undated).
- There should be no perfect plan (Weltz, undated).

There are, of course, several problems related to participation:

- Who should be included in the participatory scheme (representatives, all end-users)?
- How can participation work when one system is introduced for several thousands of employees?
- There may be overly high expectations associated with participation.
- The users do not know enough about the system to be really able to participate.
- The users are conservative and aim primarily to minimize all changes.
- Management allows only pseudo-participation without any real input from the end-users.

Consequences of Computer Use

There are good reviews on the consequences of computer use (e.g. Bjorn-Andersen and Rasmussen, 1980; Kling, 1980; Iacono and Kling, in press). In our context, three outcome variables are of particular interest: the impact of new technology (1) on social relations, (2) on control and skill utilization, and (3) on the organization.

Impact on social relations

There are very few empirical studies on this issue. However, there is some evidence for the existence of lower staff and supervisor support in computerized office work than in non-computerized work (Sauter *et al.*, 1983). Volpert (1985) and Rosenstiel (1984) argue that the introduction of computers results in increased isolation. One reason for this may be that computer systems integrate filing, getting information, deciding, and communicating. This results in less moving around, less going to the next room, etc.—all of which used to enhance informal social contacts (Salvendy, 1984; Weltz, 1982). This contrasts with informal reports that there are even stronger social ties because the computer is a constant topic of discussion. White-collar workers often do not work the whole day with computers but only some part of it (Ruch, 1986); thus, they share computers—a welcome chance for social interaction. Depending upon how the allocation of space, the division of work, and the social situation are organized, positive or negative effects on social relations are possible. However, a negative impact of computer work on the intensity of social relations is likely to happen in many organizations.

Impact on control at work and skills utilization

There is some evidence that the introduction of computers may lead to an increase in control at work and in skill utilization in the blue-collar section (Kern and Schumann, 1984; in contrast to this, see Frese and Zapf, 1987) but that there is a danger that employees' control at the white-collar level will be decreased (Buchanan and Boddy, 1982; Ellis, 1984; Hoos, 1960, 1983; Iacono and Kling, in press; Mowshowitz, 1976; Rödiger, 1985; Sauter *et al.*, 1983; Schardt and Knepel, 1981; Sydow *et al.*, 1981; however, see in contrast Reichwald, 1983, and the differentiated picture drawn by Gottschall *et al.*, 1985). Note that the prevailing tendency occurring in a particular organization is dependent upon how the work is organized (Iacono and Kling, in press).

Impact on the organization

Although it is not possible to generalize across different organizations or across different countries, some trends can be ascertained.

(1) Computers are usually introduced in an evolutionary fashion—step by step with only a few changes taking place at each point in time (Pomfrett, Olphert, and Eason, 1985, p. 847). All in all, there are fewer changes than one might assume (De Brabander *et al.*, 1981; Frese and Zapf, 1987).

(2) Any introduction of computers into the organization has some implications for the power structure of the organization (Kling, 1980) since information is redistributed and new avenues of information flow are produced. Those controlling computer resources become more powerful (Bjorn-Andersen and Rasmussen, 1980). Suggestions for or against certain systems are, therefore, often based on this power issue rather than on purely functional grounds.

(3) The advent of large mainframes in the 1960s led to an increase in centralization (Spinas, 1984); but this tendency was reduced again with the use of personal computers (Reichwald, 1983). Today, with the use of large databases and support systems for specialists (e.g. in the banking and insurance industry), there may be a tendency to recentralize. However, the meaning of centralization may become more difficult to ascertain, because there are systems that increase both the decision latitude for decentralized units and the centralization of information that gives the headquarters a great deal of knowledge on the day-to-day operations of the units (Ellis, 1984).

(4) With the fluctuation of centralization, the power of computer departments in organizations may undergo change as well.

(5) Earlier studies showed trends towards specialization when computers were introduced (partly motivated by the costs of hardware) (Bjorn-Andersen *et al.*, 1979). However, since hardware is becoming cheaper, this trend may be stopped and even reversed (Bjorn-Andersen, 1985). There are cases in the banking and insurance industry, for example, where the worker is responsible for all areas of insurance (e.g. car, life, and house insurance). This was introduced to enhance rapport with the customers (Gottschall *et al.*, 1985; Spinas, 1984).

(6) Recently, concern over invasion of privacy has become dominant—especially in some European countries and particularly in the labor unions. Bjorn-Andersen and Rasmussen (1980) report a case of a French insurance company where all the doors including the toilet door could only be opened by a plastic card being put into some terminal. Thus, management could develop a profile of movement for each person. After a strike, this project was cancelled. Computerized personnel information systems can also be used to quickly develop personal profiles of individuals since, for example, the following data are available: trends in sickness days, number of children at home, smoking and drinking (people often pay with a computerized card in the company cafeteria), trends in average keystrokes per day and which telephone numbers have been called how often. These data can then be used for lay-offs and firing, putting people on half-day jobs or realigning departments. The point is not so much that this information was not available before the advent of personnel information systems, but that the scattered data can now be more easily combined and cheaply collected and made accessible to management.

Conclusion

There is overall agreement, with just a few dissenting voices, that the so-called technological determinism (of the Blauner, 1964, variety) is wrong. There is overwhelming evidence that there is a *danger* of Taylorization of office work with the advent of computers. This could occur by bringing to the office the kind of division of labor that is boring and tedious with planning of work being placed outside of the working person's control. This is typical of much of blue-collar work today (Dieckhoff, Dieckhoff, and Roth, 1982). There is also evidence, however, that it is possible to use alternative (often sociotechnical) job design methods and, additionally, that it pays to proceed with a more holistic and humane approach (Kaye and Sutton, 1985; Margulies and Zemanek, 1983; Pava, 1983; Ranney and Carder, 1984; Ruch, 1986; Spinas, 1984, 1986; Ulich, 1981; Walton and Vittori, 1983; however, contradictory to this is Hedberg, 1980). Thus, it is not the technology that determines the human consequences of computer use, but how computer use is organized (Driscoll, 1982; Haider and Rohmert, 1983; Iacono and Kling, in press; Ulich and Troy, 1986).

Empirically, depending upon the hierarchical position of a worker and whether a worker has a data entry job, a data recall job, or a job involving an intensive dialogue with the computer (Sydow, 1984), computer use has different impacts. At the lowest level, the advent of computers created the keypunch operators who typically hold a Tayloristic job *par excellence*. Further technological development will probably reduce the number of workers in this area. At one level higher up, the introduction of word processing may downgrade the working conditions for the secretaries if a typing pool is introduced at the same time, but upgrade the typists' jobs in an already existing typing pool. At the level of specialists, decision-support systems can sometimes lead to a polarization of skills (some are downgraded, doing routine work that is essentially prescribed by the system, and some are upgraded, doing the complicated cases) or to upgrading of the job because of higher use of skills (Gottschall *et al.*, 1985). Job enrichment can be easily achieved at this level. Even at the low managerial level, as long as there is high interdependency of the systems, task constraints may develop (Bjorn-Andersen, Eason, and Robey, 1986) and there may also be tighter supervision (Wynne and Otway, 1983).

Thus, the general conclusion is that the impact of the use of computer technology is variable and depends on the type of jobs and, in the last analysis, on organizational decisions. This has optimistic implications—in so far as improvements of the working conditions are possible and can be achieved in the process of change, but it also implies a high degree of responsibility for the organization to optimize the adaptation of the technology to human needs.

STRESS

The question of whether stress at work has increased after introducing computers or decreased must be answered very similarly to the issue of organizational effects of modern technology: it depends. Thus, there are studies on people working with computers which show higher stress-effects (Frese, Saupe, and Semmer, 1981; Johnansson and Aronsson, 1984; Schardt and Knepel, 1981), lower stress-effects (Kalimo and Leppänen, 1985), and equivocal outcomes (Frese & Zapf, 1987; Sauter *et al.*, 1983; Turner and Karasek, 1984). If the job produces high stressors and little control, negative effects are to be expected.

There is now ample evidence in industrial psychology which indicates that two aspects of the workplace should be investigated under the heading of stress and strain: the stressors and the resources (Frese, in press, a). Lack of resources may have direct negative effects and/or may interact with stressors to produce negative effects, since stressors have a higher impact if resources are low. Stress-effects may manifest themselves in occupational ill-health and in disease (more on this in Cooper and Marshall, 1976). Examples of resources at the workplace are control or descretion level at work (Frese, 1984b; Frese, in press, a; Karasek, 1979; Karasek *et al.*, 1981; Semmer, 1984), amount of skill utilization (Karasek, 1979), and social support (House, 1981; Frese, in preparation b). Therefore, two questions are to be dealt with: (1) Are there new (computer-related) stressors at work? (2) Are there new (computer-related) resources or threats to resources?

New Stressors in Computer-aided Work?

Physical stressors

A major part of research on potential negative effects of the use of a video display terminal (VDT) has been directed to the study of posture effects, radiation effects, and medical effects on the visual system. This literature is not reviewed here. However, it is plausible that purely physical stressors can feed into psychosomatic problems. There is evidence that VDT work is physiologically demanding on the eye (Haider, Kundi, and Weissenböck, 1980; Stark and Johnston, 1984; Smith, 1984; Wilkins, 1985). However, this does not necessarily produce *eye-strain*: sometimes it does (Gunnarson, 1984), sometimes it does not (Hartmann and Zwahlen, 1985; Howarth and Istance, 1985). Obviously, this depends on how badly or how well the VDT is designed from a hardware ergonomics point of view (Stammerjohn, Smith, and Cohen, 1981; cf. Cako *et al.*'s 1979 discussion of this). But there is an additional and not quite so obvious point: eye-strain is not only a function of the specific demand on the eye but also of general stress at work. There are three psychosomatic processes by which general stress at work may lead to eye-strain (cf. Frese, in press, b, for a similar reasoning for musculoskeletal complaints): (1) Every stressor demands some coping attempt. Since humans have only a limited processing capacity, stressors reduce the available processing capacity that is left for the original task. Thus, the task becomes objectively more difficult (this also explains a lower performance rate under stress; Dainoff, 1984). (2) When there are high background psychological stressors, higher muscular tension ensues as a general psychological response; this also affects eye fatigue. (3) High background stress leads to a stronger pain sensation.

The psychosomatic reactions most easily appear in that organ that is most strongly in demand and strained in a physiological sense. Since general stress at work has an impact on eye-strain in jobs that put an emphasis on work with the eye, this reasoning allows us to interpret the finding that there is an association of eye-strain with time spent on a VDT in 'bad' jobs (like data entry by typists who have high stressors and little control) but not in 'good' jobs (like programmers or secretaries with a wide range of tasks) (e.g. Coe *et al.*, 1980; Dainoff, *et al.*, 1981; Läubli *et al.*, 1980; Smith, 1984; Smith *et al.*, 1981; Cakir, 1981). Apparently, additional psychological stressors in 'bad' jobs contribute to eye-strain.

Thus, there is evidence that VDT work is psychologically demanding on the eyes and that this *plus* a workplace with little control and high stressors can lead to eye-strain. Other (psycho)somatic reactions to the VDT workplace have been described as headaches and backaches (Cakir, 1981). A similar reasoning might apply here, although the latter may be a function of bad ergonomic design that forces the worker to lean forward, thus not having a support of a chairback.

Psychological stressors

It is necessary to ask whether there are new psychological stressors related to working with a computer and/or whether some well-known stressors become more pronounced in computer work. There are five stress problems that fit into that category: response time and computer breakdowns, feeling rushed, constantly feeling supervised, invasion of privacy, and the abstractness of computer work. A sixth stressor is the threat of unemployment which is not new but has become increasingly important with the advent of a powerful new technology.

System response times and breakdowns have been shown to be a major stressor in computer-related work (Johansson and Aronsson, 1984). (They are, of course, not entirely new to computer systems. Machine breakdowns or organizational problems have been disturbing in non-computerized industry, as well; Semmer, 1982). In a way, it is paradoxical that slow system response times of the computer and breakdowns should count as a stressor. One might think that they would be welcomed as beneficial rest periods (or thinking periods). The stress-effect has been explained as being due to unpredictability of events (Boucsein, Greif, and Wittekamp, 1984) or it may be due to the interruption of a plan. Miller (1968) and Cakir (1986) gave recommendations on response times. In general, low response times have been suggested. However, Shneiderman (1986) warned that very short response times may also add to feelings of being rushed—as a further stressor. In fact, research has not been entirely consistent in showing the lowest stress-effects with low response times (Kuhmann *et al.*, in press; Martin and Corl, 1986; Schaefer *et al.*, 1986). However, the question arises whether laboratory studies, like the ones by Kuhmann *et al.* and Schaefer *et al.*, where response times of 8 sec were actually overall less stressful than those of 2 sec, can simulate the time pressures that exist in normal work which may be the most important contributor to the annoyance with long system response times.

In a thorough review, Shneiderman (1984b) argues that it is necessary to differentiate between system response time and thinking time. When the system response time becomes much longer than thinking time, it becomes a stressor. The stress characteristic of system response time is also dependent on what the user expects to be a normal response time. This may be one factor explaining why high variability of the response times is most often reported as being stressful. Stable expectations cannot be developed in this case.

The effects of breakdowns and response times will most probably also depend on organizational issues, namely how the workers are paid (e.g. some piece rate system), whether they are under public pressure to render results (e.g. in an airline ticket office), or whether or not their workload is adjusted to breakdowns or slow system response times.

Thus, the aversiveness of response times may be a function of being rushed. Johansson and Aronsson (1984) and Weltz (1982) have argued that the *feeling of being rushed and the feeling of high concentration* are produced by working with a VDT. The cursor constantly blinks, signalling readiness for further input. The typewriter used to provide for a mixture of high-concentration tasks like writing and low-concentation tasks like getting up, fetching folders, and putting paper in the machine. This mixture of tasks also made it possible to rest for very short (nearly not noticeable) periods of time. These tasks are now streamlined into the system. One user described the work with the VDT as 'simpler, yes, but somehow more strenuous' (Spinas, 1986, p. 12). Again, these feelings may gain added importance because rushing people used to be done by supervisors in the office (if at all) and not by machines. Now it is more difficult to consciously ward off the time pressure.

There is new potential to *supervise* white-collar workers. Many computer programs provide a convenient log of how many keystrokes a worker has made per day. This kind of information creates a dehumanizing atmosphere (Smith, 1984) and is probably one of the reasons why anxiety is often quite high when computers are introduced in the office.

A similar issue is *invasion of privacy* which is surprisingly little researched. Computers allow for the easy combination of data that are, singly and taken individually, relatively innocuous. With the use of computers for registering citizens, keeping track of their financial situation and prosecuting criminals, the danger of invasion of privacy in society is widely feared (Rosenstiel, 1984). Similar fears come up in industry, as well, when many work-related and private data are stored and combined.

Abstractness of work may be another potential stressor (Volpert, 1985; Weltz, 1982) although it has not been researched much. This may be a more important problem in the blue-collar sector (e.g. in the printing industry where concrete handling of lead letters was replaced by computerized type-setting). But even in white-collar work, people used to handle paper, folders, and numbers in a relatively concrete way. There was a one-to-one relationship between a piece of paper and the information on it. This relationship is gone in computer-aided work. Abstractness may be of added importance in telecommunication (which we do not discuss here). There may be a 'natural' need in humans to handle things in a concrete manner (a 'full action' means that both thinking and handling something are combined; see Hacker, 1985). This may also be a reason why 'direct manipulation' is a successful interaction mode. Lack of concreteness may lead to two problems: (1) a feeling of unreality towards the objects—this might have more negative consequences in war games than in office work; (2) a fear of making errors because the errors are made

with a medium that is not discernible. An example that was reported to me in a company might illustrate this. When a data system was introduced one secretary kept a column of names constant and by mistake changed their addresses around in the second column. Such an error is, of course, impossible to make on paper.

The last stress factor to be mentioned here is *fear of unemployment*. While simulation studies of national economies have given equivocal results on the problem of unemployment, there is no doubt that many companies introduce computers to reduce the number of workers. This leads to fear of unemployment, which in turn leads to high stress-effects (Pelzmann, Winkler, and Zewell, 1985).

Does this list of stressors imply that computers have led to a higher degree of overall stress at work? I do not think that this conclusion is warranted. Some stressors have been minimized, e.g. anxieties about making a 'typo', or having to retype texts over and over again after small changes. Additionally, computers can be used to reduce the demands on working memory, another potential stressor (Hacker, 1983). It depends on the organization of work as to whether the net stress-effect has increased and the net resources have reduced.

New Threats to Resources in Computer-aided Work

There are four major resources at work: control at work, development of skills, information support, and social support. It has already been stated that there is a tendency to reduce *control* in the computerized office. Control *vis-à-vis* the system may mean to have options, to be able to adjust computer programs to one's own job requirements and to one's liking, to develop one's own masks (screen layouts) or patterns of interaction, to be able to choose the tasks and the task order, and to time them at one's own pace—in short, to be master of the system. As already stated, control depends on the system and on the organizational context. Therefore, it is not surprising to find different results on control perception in computer-aided work (Buchanan and Boddy, 1982; Rafaeli and Sutton, 1986; Sauter *et al.*, 1983; Turner and Karasek, 1984). However, there is no doubt that control decreases strain and increases job satisfaction. Indirect or direct evidence for this can be found in Bikson and Gutek (1983), Smith *et al.* (1981), Troy (1986), Turner and Karasek (1984), and Ulich (1986).

Without *skills*, there is little chance to master the machine. Skills can be used to change the program, to adapt it to one's needs, and to develop clever solutions for complicated problems. Thus, skills can help to reduce the effects of stress (Greif, 1986b). One prerequisite for skill utilization is that work is complex enough (the relationship between control and complexity is itself quite complicated and cannot be pursued here; see Frese, in press, a). Hacker (1983, 1985) has shown that complexity of work leads to positive consequences. However, this is only true given a certain skill level. Another prerequisite is that skills are to a certain extent private. This means that it is up to the workers to decide to employ them. If they are not private, a skillful problem solution might be programmed into the system. In this case, the skillful accomplishment is not at the worker's disposal any more, but is a resource of the machine. Thus, it cannot count as a resource for the individual's stress management because it does not increase a sense of self-confidence and it cannot be used to reduce unexpected stressors. On the contrary, in such a case the job has been reduced to a potentially more monotonous work situation. Thus, skill utilization by the workers should be kept (and enhanced) in computer-aided work. This may not be a problem at the present time because the systems are nowadays still complex enough to actually necessitate an increase in skills. But with increasing employment of 'intelligent' systems in the future there is a danger that skill utilization will be decreased.

One frequent phenomenon in computerized office systems is that units of *information* are hierarchically organized according to their access so that one needs to know certain codes in order to get information that is reserved for those higher up in the hierarchy. Since official task descriptions are seldom complete or even adequate, and since they determine the information policy of the system, this leads to a person not having the needed information. Since one reason for regulating access to information may be the privacy issue, there might be a trade-off here between the problem of invasion of privacy and the question of information support.

Social support is a resource that buffers the effects of stressors on psychological and psychosomatic problems (House, 1981). Since social support is dependent on social contacts, the frequently mentioned problem of social isolation as a result of computer-aided work is important. Computer-aided work, e.g. for the specialist, can lead to less social contact (Smith, 1984; Turner and Karasek, 1984). In the past, the specialist used to move around and ask other people for information. This may not be necessary after a system has been introduced. However, to my knowledge there is little systematic research relating this problem to the stress field.

In summary, there are dangers of increasing stressors as well as reducing resources in computer-aided work. However, this is dependent on the particular organizational context in which the work is done. As a matter of fact, resources can be increased and stressors decreased with the introduction of computers in the workplace as well. Resources are usually enhanced when some kind of mixed work is instituted, e.g. when a secretary not only types but also does scheduling and financing, writes letters on her own, etc. (Ruch, 1986; Troy, 1986; Ulich and Troy, 1986).

Our emphasis on organizational conditions leads to the general warning that even the best designed computer screens and keyboards (from a hardware ergonomics point of view), as well as the best software ergonomics, do not help if the organizational environment produces stressors and decreases resources (see Hacker, 1983, Rödiger, 1985).

COGNITIVE OPTIMIZATION OF HUMAN-COMPUTER INTERACTION

Cognitive optimization of human-computer interaction can be done by increasing the individual's skills to deal with the system and/or by optimizing the system. Both increasing skills and optimizing systems are related to the question of how a human makes sense of the system—the mental model. Training is a direct way of teaching the 'sense' of the system; but designing the system should similarly be oriented to implicit or explicit mental models. Therefore, I shall first discuss the concept of mental model, then the optimization of skills (training), and finally, optimization of the system.

Mental Models

It is nowadays nearly a truism that a person using a computer has to develop some kind of mental model of the functioning of the system and that even a novice will approach a computer with some kind of mental model in mind (overviews: Carroll, 1984; Jagodzinski, 1983; Rohr and Tauber, 1984). In other words, the user has some kind of conceptualization of the functions of the system and of how one has to deal with the system. These mental models may be of the metaphor kind, e.g. using the analogy of a typewriter when first using a word-processing system, or may be a knowledge of the rules to be used in a system. It is useful to make the mental model explicit for the user (Kieras and Bovair, 1984) because wrong mental models may lead to wrong or inefficient approaches to the machine (Bayman and Mayer, 1984). However, this is not always so; in certain circumstances a wrong mental model may be more functional than a correct one that is just a little incomplete: a wrong home heat regulation theory (valve theory) saves more energy than the correct feedback theory which does not take into consideration other thermal laws (Kempton, 1986). Norman (1983a) observes the following characteristics of mental models: They are incomplete, they are not 'run' easily, they are unstable (and parts of them forgotten), they have no firm boundaries, they are unscientific (and even superstitious), and they are parsimonious. In short, they are not in any way a neat and non-contradictable set of ideas as in a scientific model.

Essentially, the following conceptualizations of mental models in research on human-computer interaction can be differentiated: (1) the mental model may refer to action or to knowledge of the world; (2) the mental model may or may not be thought of as consisting of hierarchies; (3) the mental model may be of the production theory type or be more holistic, like the schema theories; and (4) the mental model may be analog or analytic.

(1) Various theories have distinguished action-oriented models from models that explain and describe the world. Miller, Galanter, and Pribram (1960) distinguished image (the description) from plans (the action-oriented model), Hacker (1978) an operative image system from other non-operative cognitions, Young (1983) a surrogate model from a task/action model, Anderson (1983) declarative from procedural knowledge, Carroll (1984) conceptual from mapping analysis, and diSassa (1986) structural from functional models. Not all of these distinctions are exactly alike but most agree that there are mental structures for description and/or analysis as well as mental structures for action. The relationship between these two different kinds of mental models still needs to be elaborated: it is likely that in the last analysis the action-oriented models are the important ones and the conceptual and analytic ones are subservient (Hacker, 1978). This may be one of the reasons why functional models are more useful in learning about computer systems (diSassa, 1986).

(2) The mental model can be conceptualized as being hierarchically organized (e.g. Anderson, 1983; Gallistel, 1980; Miller, Galanter, and Pribram, 1960) or not (Neisser, 1976). Most cognitive theorists assume now that there is some kind of hierarchy, but that processing may also run counter to a strict hierarchial model (heterarchical conceptualization; Gallistel, 1980).

(3) Many theorists in the area of human-computer interaction think of the mental model as consisting of production rules (Anderson, 1983; Card *et al.*, 1983; Polson

and Kieras, 1985). The alternative would assume a less elementaristic notion, as in the concept of schema (Bartlett, 1932; Neisser,1985). The production system consists of low-level molecular production rules (e.g. given a certain goal and certain conditions, act X should be performed). In contrast, schemata are more high-level molar concepts—often thought to have some *Gestalt* character—that cannot be broken down into smaller units. Schemata are flexible, anticipatory devices that can have internal conceptualizations of movements and trajectories (Neisser, 1985) often related to images. An integration of both views has been suggested (Waern, 1985): high-level schemata are used in novel situations or in non-redundant environments, while low-level automatisms of the production system kind are used in overlearned skills. In fact, lower-level skills are more easily disturbed, either by thoughts (in the fable the centipede cannot walk any more after being asked how he moves his hundred legs) or by novel inputs from the environment, while the higher regulation level can easily deal with novel input (at the expense of being less efficient). This integrative view has to answer the difficult question, of course, of by which process a high-level schema can be transformed into a low-level production system in the process of learning an action.

Elementaristic production systems can explain the process of automation easier (through the chunking process) than schema theories (Anderson, 1983). On the other hand, schema theories are better equipped to deal with metaphors, images, the incompleteness and flexibility of mental models, the active, changing, and constructive nature of mental models, and their *Gestalt*-like nature.

(4) The mental models may be either analog or analytic. This is somewhat related to the above point. While images and schemata are typically analog mental models, production rules are more analytic. Metaphors (Carroll and Thomas, 1982) play a role when learning a computer system. The typewriter metaphor is typically used by novices when approaching a word-processing system. Metaphors are analog mental models. It has been suggested that novices start with models of an analog type and become more analytic with time.

Hammond and Barnard (1984, p. 131) specify in some more detail what the user needs to know in order to work with a computer system: 'Knowledge of the domain', 'Knowledge of the computer version of the domain', 'Knowledge of the workbase version of the domain', 'Knowledge of the problem', 'Knowledge of system operations', 'Knowledge of physical interface', 'Knowledge of interface dialogue', 'Knowledge of natural language', and 'Knowledge of other machines and procedures'. This list is quite impressive, and other authors, like Greif (1986b), add that one should also know the potential errors. Riley (1986) contends that a user should have an understanding that is internally coherent, valid, and integrated.

Seen from this perspective, it is not surprising that people make serious errors with computers and that learning takes a long time. Additionally, this complex set of issues makes the design of software quite complicated. In any case a system designer has to take into consideration the users' mental model—otherwise the system may not be functional and usable.

Optimizing Skills: Training

In order to develop a mental model, training is necessary (although not all training is geared towards developing mental models). But training does not only optimize the

interaction with the system but can also decrease potential stress-effects. Workers who are not well-trained use descriptors for the computer that signify that they see themselves as servants of the system ('he wants to') in contrast to the well-trained workers (Ulich and Troy, 1986). Although, in general, organizations perceive the need for the training of skills in the area of human-computer interaction, they usually do not provide enough time and resources (Bjorn-Andersen, 1985; Gottschall *et al.*, 1985). Training research has concentrated on programming (which will not be dealt with here; see Boulay and O'Shea, 1981; Mayer, 1975, 1976; Owen, 1986) and word processing on which substantial work has been done at the IBM Watson Research Center (e.g. Carroll *et al.*, 1985; Carroll and Mack, 1984; Mack, Lewis, and Carroll, 1983).

The less complicated the commands, the rules, and the interface of a system, and the more the person knows right from the start, the less there is a need for training. However, even for 'fool-proof' systems, such as Apple's LISA with its desktop metaphor, some type of training will be needed. They are difficult to learn, even for persons who know other word-processing systems (Carroll and Mazur, 1985). Training is necessary even if so-called on-line tutorials or manuals are provided. Manuals are not used much (Carroll and Mack, 1983; Scharer, 1983) and tutorials are often not helpful (Carroll and Mazur, 1985; Greif, 1986a).

Most training programs can be conveniently grouped as in Table 5.1 (although there are really never 'pure' types of training program). Sequential training programs do not explain the background of the system and the laws and rules regulating it, but essentially present a correct sequence of steps and have the student practice them. Thus, they work within a behavioristic tradition. Most computer-driven tutorials follow this kind of reasoning, presenting a step and then asking the person to perform the respective action (Carroll and Mazur, 1985; Greif, 1986a). Unfortunately, many commercially available training programs do the same (Greif, 1986b). In contrast, an integrated-systematic training program explains the background for the computer program, the reasons behind the commands, how they are related, which metaphors are used, how the commands pertain to general rules and heuristics—in short, to some mental model. These explanations are action-oriented. In other words, an integrated-systematic program tries to build up complete and full action. This implies that all levels of regulation are implicated in the action: the intellectual level, on which rules, metaphors, and heuristics are regulated, as well as the lower levels, e.g. the sensorimotor level of regulation (cf. Hacker, 1985). In a sequential program, only the lower levels of regulation are trained (however, the trainee may well develop some kind of action-oriented and even systematic mental model in spite of the training program). Thus, integrated-systematic training stems

TABLE 5.1—Types of Training Programs

		TRAINING PROCESS	
		SEQUENTIAL	INTEGRATED-SYSTEMATIC
Development of the Mental Model	Passive		
	Active		

from an implicit or explicit theory of action (Norman, 1986a; Frese and Sabini, 1985; Volpert, 1981). In general, integrated-systematic forms of training seem to work better than sequential programs (Frese *et al.*, in preparation; Greif, 1986a; Hacker, Rühle, and Schneider, 1978; Kieras and Bovair, 1984; Volpert, Frommann, and Munzert, 1984; a good description of such a training is given in Greif, 1986b). Moreover, integrated-systematic training is better adapted to how people go about learning; humans give spontaneous interpretations, generalize from experience (even when they refer to very shaky data), and perform their first steps on the computer in accordance with some preestablished metaphors (Carroll and Mack, 1984; Douglas and Moran, 1983). Usually, computer-naïve people take a typewriting metaphor when they work with a word-processing system and make the 'appropriate' mistakes (Mack, 1984; Waern, 1985).

The second dimension of Table 5.1, 'active-passive', is related to an action theory approach as well. Carroll and Mack (1984) vividly describe how trainees proceed *actively* and exploringly with a word-processing task rather than passively following instructions (as some manuals would presuppose) (cf. also Waern and Rabenius, 1985). Training that emphasizes exploratory behavior fares better than more traditional procedures (Carroll *et al.*, 1985; Greif, 1986a; Frese *et al.*, 1986; cf. Carroll and Rosson, in press, for a theoretical discussion on this issue). Unfortunately, it is nearly impossible to neatly differentiate between the four cells of Table 5.1; it is particularly difficult to develop an active training program that is not integrated-systematic since people spontaneously develop models with some degree of integration. Frese *et al.* (1986) have tried to test three of the four cells against each other with the active and integrated-systematic training being the most effective cell.

There are five more issues in training: (1) Training of what? (2) Treatment of errors. (3) Transfer. (4) Local experts. (5) Individual differences.

(1) Training should not just deal with those few tasks that appear in a particular job description but should be broader so that the user really understands the system. Riley (1986) has suggested that user understanding of a system implies a conceptualization with some internal coherence, a correct representation of the system, and an integration of this knowledge into other areas of knowledge. This principle of broad training objectives is particularly important for computer-based work because even well-established software programs often have serious flaws. To deal with these flaws, one has to know a great deal more than is necessary to do just the task in hand. Furthermore, the official job description seldom represents all the tasks that a worker really does. Bjorn-Andersen (1985) suggests that training should also include democratic participatory and exploratory behavior at the workplace.

(2) Making errors produces anxiety, particularly in the novice. There are essentially three suggestions for dealing with them. First, the training program does not allow errors or minimizes the chances to make errors, as in the no-surprise editor (Mack, 1985) or in the training wheels approach (Carroll and Carrithers, 1984). Second, the system facilitates retrieval from error, e.g. with an 'undo' button (Carroll *et al.*, 1985). Third, trainees are encouraged to make errors (a sort of error training) and errors are used in a specific procedure to develop a mental model of the system (Greif, 1986a, 1986b).

(3) Since it is nearly certain that workers will have to use different systems in the course of their working life, the question of transfer between systems is important. Carroll and Mazur (1985) report how people have many problems with the supposedly easy direct manipulation interface of LISA. Since all of the subjects in this study had known other systems before, some of the problems might actually have been instances of negative transfer. Examples of the difficulties of transfer are also given in Karat *et al.* (1986). This is an area with relatively little research. Research is particularly needed for answering the question of which *basic* skills a person should acquire to deal with *different* kinds of programs. An additional issue is to transfer the skills learned in some course to one's daily work. This is a particular problem with courses that are not task-oriented and that take place outside work. Therefore, training should be oriented to the concrete work that has to be done, transfer tasks should be given so that the workers can solve their daily work problems, and check-ups and additional 'refreshing-the-memory' training should be given.

(4) In most companies some people will become the explicit or implicit local experts for given departments (Scharer, 1983). A system of local experts should be developed by the company because people like to ask other people about the procedures to be taken and because the local expert can stimulate further learning of the system and support the long-range learning effects.

(5) There may be individual differences in the ways people learn. For example, people with low spatial memory are poor in learning line as well as full-screen editors (Gomez and Egan, 1983). People may even differ in their preference for either of the two dimensions of training, stated above. There may be learning strategies that are sequential or integrated-systematic (Veer and Beishuizen, undated). Similarly, some people like just to be active (without reading a lot beforehand) and others want to read a good deal before they start to learn a word-processing system (Frese *et al.*, 1986; Schulte-Göcking, 1987).

In conclusion, there is evidence that training should be action-oriented and should teach an explicit mental model, the training goals should be broader than just teaching the officially described tasks, errors should be minimized in the training process or explicitly taught, overcoming problems of transfer should be an integral part of training, local experts should be encouraged, and individual differences should be taken into consideration.

Optimizing the System

The optimization of system parameters has been the research area to which scientists and human factor workers have contributed most actively. It cannot and shall not be completely summarized here. Obviously, a system has to be compatible with prior mental models, it has to present an obvious system's image and has to be self-explanatory, it has to give clear feedback and decrease mental overload to be useful for the development of an adequate mental model. Furthermore, it has to have a clear layout or mask (Morland, 1983). I shall pick out some issues for discussion here that seem to be of major (and increasing) importance: help facilities, different types of interaction modes, and errors.

Help facilities

Manuals and on-line help systems are usually constructed for the novice. Nevertheless, novices do not use them under 'normal' working conditions (expert users refer to them more often) (Mack, Lewis, and Carroll, 1983; Scharer, 1983). This is not surprising because 'to ask a question, one must know enough to know what is not known' (Miyake and Norman, 1979). It is often the expert's task to help users develop the right questions (Pollak, 1985; McKendree and Carroll, 1986).

Novices frequently do not understand computer jargon, do not know how to get to the appropriate help message, or do not want to plow through a lot of information. Users like to ask their peers for advice (Lang, Auld, and Lang, 1982; O'Malley, 1986); therefore many offices have 'local experts' who have the (sometimes informal) function of providing human help (Bannon, 1986; Scharer, 1983).

Manuals Manuals are not read, warnings are not heeded, even clear recommendations by the computer are sometimes not understood. Sullivan and Chapanis (1983) suggest the following rules for writing manuals: simple language, short and active sentences, order of description parallel to action steps, complete and specific description of action, one thing at a time, headings and subheadings, lists instead of prose. Carroll *et al.* (1986) have suggested a more radical approach. They constructed a 'minimal manual' with a minimum of words, which is directed towards error recognition and error recovery and which is task-oriented. It presupposes that the user will only consult the manual as a starting point. Compared with a commercially available longer manual, this minimal manual leads to better performance, although it does not give advance organizers (Foss, Rosson, and Smith, 1982), it is not complete, and it does not have a clear hierarchical outline.

On-line help systems O'Malley *et al.* (1983) distinguish the following user needs with regard to help functions: quick reference (e.g. verification of a command name), task-specific help (the user knows the problem but not the command for solving it), and full explanation of the capabilities of the system. By and large, on-line help systems usually deal only with the first two needs.

Help facilities may be either passive (the user has to call upon it) or active (the help is initiated and/or selected by the system). Cohill and Williges (1985) studied the effects of help being initiated and selected either by the user or by the system, and whether the help was presented on-line or as hard copy. The best condition was the user initiated and selected help that was printed on hard copy.

Depending upon design, on-line help systems may either be the same for everybody or differ according to the expert level of the user. Help systems that adjust to the user level of expertise can be designed as knowledge-based help systems (Fischer *et al.*, 1985; Dzida, Hoffmann, and Valder, 1984). These help systems may sometimes have the form of excursions that do not change the state of task operations (Darlington, Dzida, and Herda, 1983). One prerequisite for knowledge-based help systems is to develop some kind of user model for the system. Chin (1986) describes an example of how to do this: the system records errors and the difficulty levels of commands used correctly and incorrectly. The records are then compared with some preestablished pattern of errors and command knowledge stored in the system. Thus, the system has an *a priori* categorization for typical errors and levels of competence of novices,

beginners, intermediates, and experts. On the basis of this comparison, the system assigns the user to one of these categories and then gives appropriate help (e.g. more explanations to the novice and just a shorthand description to the intermediate (cf. also Williges *et al.*, 1985).

Interaction mode

Usually, users develop their mental models from interacting with a system. Thus, the surface structure of the system is important. There are different modes by which a person can interact with a computer. These interaction modes may be based on menu, command language, natural language, and direct manipulation.

Menu vs. command vs. natural language Conventionally, menu, command language, and natural language systems are differentiated. Menus display the different commands that can be used. They can either pop up when ticked (as in the Macintosh system) or may be displayed more permanently. They can also be embedded in a text which is particularly useful in data systems (Koved and Shneiderman, 1986). They can display icons or commands with a short explanation. Finally, the menus may just display commands and thus help the user to remember the commands. The menu contents may be touched with the help of a mouse, the finger, or a light-pen, or activated by moving the cursor with the keyboard. From a memory point of view, menus allow for recognition rather than for recall which is why menus are easier to learn.

Command language systems, on the other hand, rely on recall. In a pure command language system, the user has to know the different commands (thus, it is similar to a programming language even if just used for word processing). However, these systems are often complemented by named function keys which actually act as menus (this time presented on the keyboard instead of on the screen).

A natural language system is based on our own natural language (e.g. English). We tell the computer in our language what we want it to do. Unfortunately, there are no systems that really understand natural language in a 'natural' way. Thus, the systems that exist are relatively restricted (e.g. they are not able to understand when there is a small typing error in the natural language command). Therefore, it is quite difficult to test the usefulness of different natural language systems. Small and Weldon (1983) used a trick to enable to compare a real natural language system with a command-based system in a query task. In the natural language condition, the user's requests were displayed to another (hidden) human experimenter who interpreted the questions, reacted to the requests, and gave the appropriate answers. Surprisingly, the natural language system did not fare very well. Although it was not less accurate than the command system, it produced slower results.

This may reinforce the skepticism towards natural language approaches. Shneiderman (1980) argues accordingly that constrained natural language systems make for negative transfer from real natural language, that the differences between computers and people get confused, that natural language systems are unreliable and will always be so. The negative transfer effect of natural language commands was observed by Scapin (1981), as well.

Other empirical studies comparing performance with command, menu, and natural language systems do not render entirely consistent results. Hauptmann and Green

(1983) conclude that there are few differences between these systems in the time it takes to operate and in satisfaction because the structure and constraints of the programs wipe out all other effects. The natural language group was most annoyed by the constraints of the commands that they could use. (Examples of how to overcome the problems of constraints in natural language programs are given by Hayes and Reddy, 1983.) Natural language also showed most verbosity (that did not translate into slower work in this study, however).

Comparing command and menu systems that are on the market, Whiteside *et al.* (1985) found few differences between them. Although their best system turned out to be a command system, other command systems were doing rather badly. Users' attitudes (satisfaction) are in general not a good indicator of performance (Barnard *et al.*, 1981; Frese *et al.*, 1986)—this was also true in the Whiteside *et al.* study.

It has been argued that menu systems are better for beginners and command language systems better for experts (Nickerson, 1986). However, contrary to this popular wisdom, there is evidence that those systems that are easier to learn are also easier to use for both experts and novices alike (Roberts and Moran, 1983; Whiteside *et al.*, 1985; cf. Stelovsky and Sugaya, 1985, and Norman, 1983b, for the pros and cons of menus and command language systems).

A special topic in command language: naming Two questions are particularly important in designing command languages: how the commands should be named and how they should be abbreviated. (They are, of course, not only relevant in connection with command languages but are especially important here.)

As Carroll (1982) argues and shows empirically, names are never completely arbitrary entities but are paradigmatic—only a certain range of names is possible and a specific name can only stand for a restricted number of objects (or processes). When observing naming behavior for files, 85 per cent were organized into paradigms (like Text1, Text2, etc.). In experiments, congruent names for commands (e.g. up/down, raise/lower, but not up/lower) are better remembered and used more efficiently in problem-solving tasks. The data for 'hierarchicalness' of names are less clear but this issue seems to be less important. Similarly, the 'suggestiveness' of command names plays a crucial role in learning an internal model and for performance accuracy (Rosenberg, 1982).

There are two controversies regarding command names that have not yet been fully resolved: are general terms better than specific ones, and are self-generated terms and abbreviations better than other-generated ones? First, Barnard *et al.* (1982) showed that specific command names were more efficiently used and produced better memory effects (albeit not all of these differences were significant) than more general names (see also Rosenberg and Moran, 1985, on this topic). Scapin (1982), on the other hand, found the opposite effect.

One of the factors responsible for this contradiction may be the question of self-generation of names. Scapin (1982) used self-generated names, in contrast to the imposed command names used by Barnard *et al.* (1982). It might be assumed that self-generated names are more easily remembered. However, there are some experiments that show the opposite effect (e.g. Grudin and Barnard, 1985, for abbreviating words). Jones and Landauer (1985) may have resolved this issue: they argue and show experimentally that the advantage of self-generation of names is lost when the subjects do not know anything about the context. Thus, the best naming is done by the person who knows the context and uses a strategy of congruence and consistency. General commands may be better learned under these conditions as well.

A special topic in menu systems: breadth vs. depth A crucial question is how to arrange a menu. The literature converges on the suggestion that the depth of a menu should not be large (i.e. there should only be a few levels in the menu tree). This speaks for broad menus (Kiger, 1984; Landauer and Nachbar, 1985; Tullis, 1985). But could it be that there is an optimal breadth of menu? Landauer and Nachbar's and Tullis's studies propose that the broader the menu, the better. Kiger's research, on the other hand, calls for an optimality criterion; the breadth should be around eight entries; this is in accord with G. A. Miller's (1956) estimate of the short-term memory capacity.

Direct manipulation Shneiderman (1982b, 1983a) coined this term to mean a system that continuously represents the object of interest, in which a complex syntax is replaced by physical action, and which makes the impact of incremental operations immediately visible. In other words, 'what you see is what you get'. In contrast to, for example, a command language which gives an abstract representation of the tasks, direct manipulation models the world that a person works on (Hutchins *et al.*, 1986). An example (albeit not necessarily a perfect one) of direct manipulation is the desktop metaphor of the Xerox Star System and Macintosh. A better example is the direct manipulation of data (Hutchins *et al.*, 1986): when a graphic shows that there are two distinct subgroups in a datapool, one subgroup is circled and it, as well as the respective statistics (e.g. the correlation for this subgroup), becomes visible in a second window.

The advantages of direct manipulation are that novices can learn the functions quickly, that work is rapid, error messages are rarely needed, users see immediately whether an action leads to a goal, actions are reversible at any time, and the system is comprehensible (Shneiderman, 1982b). Therefore, the operator feels in control at all times.

A direct manipulation interface has some drawbacks as well. For example, a repetitive operation can be done more easily with a formula (symbolic description), accuracy may be more difficult to achieve with direct manipulation devices, and finally, direct manipulation interfaces give control over objects of goals but not over the program of the computer (Hutchins *et al.*, 1986).

As the Macintosh programs are usually considered to come close to the concept of direct manipulation (e.g. by Fähnrich and Ziegler, 1985), the question arises whether they lead to better performance than, for example, menu systems. This is generally assumed but not frequently shown. The evidence for the superiority of the desktop metaphor is not unequivocal (Dumais and Jones, 1985; Whiteside *et al.*, 1985).

Direct manipulation often implies graphic representations (but note that a full-screen editor is also more direct than a line editor and that graphic *representation* does not necessarily imply direct manipulation. An overview of graphic representation is given by Gorny (1984). In general, graphic representations are of value, as long as a real-world model of the task is useful and functional, and the user is accustomed to this kind of representation (Boecker, Fischer, and Nieper, 1986; Cole, 1986; Powers *et al.*, 1984; Rohr, 1984; Widdel and Kaster, 1985; see also Preece's, 1983, critical discussion).

Errors and treatment of errors

Within his action-theoretic approach, Norman (1984b) distinguishes slips from mistakes. Slips are inappropriate actions, where the intention was correct (example: a person inadvertently deletes a whole file without wanting to). Slips can occur because of faulty activation of schemata (an example is the capture error: a person who wants to change his coat in the bedroom, undresses and goes to bed) and from faulty triggering of active schemata (Norman, 1981).

Mistakes are caused by inappropriate intentions (example: a person wanted to delete a file and finds out afterwards that she still needs it). The concept of mistakes is related to inefficiency of action (see Schönpflug, 1985; Semmer and Frese, 1985), since we often label something as a mistake when the goal is not achieved as fast and with as little effort as would have been possible. Another aspect of mistake is misdiagnosis that is enhanced by a tendency to search only for confirming evidence and to use partial explanations (Norman, 1984b).

In human-computer interaction, most 'errors' are not really human errors at all but are due to the inability of the computer program to decipher unclear commands (Lewis and Norman, 1986). To make matters worse, many programs give error messages that the user is not able to understand (examples in Lewis and Norman, 1986). There are two ways to deal with human 'errors':

1. Avoid errors. Direct manipulation is one way to avoid errors. It is also possible to program for errors, e.g. small misspellings of commands are 'understood' by the system. Additionally, a system may facilitate the retrieval of a document name by allowing the user to recall those parts of the name that he still remembers—e.g. when he had stored it, when he last worked on it—or the approximate name (Branscomb and Thomas, 1984). If the system gives clear feedback (e.g. in which mode a user is working and is consistent, fewer errors develop (Lewis and Norman, 1986).
2. Give adequate feedback when errors have happened. Dean (1982) and Shneiderman (1982a) give recommendations (see also Isa *et al.*, 1983). At the very least, error messages should be polite and constructive, specific and oriented towards the user.

INDIVIDUAL DIFFERENCES

Although we can be quite confident that individual differences play a role in human-computer interaction, there is yet little systematic research. Therefore, we cannot tell with certainty which person variables are particularly important (Muylwijk, Veer, and Waern, 1983; Veer *et al.*, 198). Likely candidates are cognitive styles (Robertson, 1985), action styles (Frese, Stewart, and Hannover, in press), and learning styles. For example, one would hypothesize that the kinds of errors made by impulsive workers (errors of commision) would be different from those made by reflective workers (errors

of omission). An impulsive worker may need an 'undo' command more often than a reflective worker. Similarly, the action styles of planfulness and goal-orientation (Frese *et al.*, in press) might influence which strategy is preferred. A highly planful person wants to lay out the plan of work beforehand, a person with low planfulness will start right away, completing the plan as he goes; differences in preferences for planning tools will depend upon planfulness. Finally, learning strategies can be more holistic or more serialistic (Pask and Scott, 1972), or can be oriented towards learning by doing (without looking at a manual) vs. learning by studying manuals first (Frese *et al.*, 1986; Schulte-Göcking, 1987). It has been argued that, in a way, these different strategies and styles call for differences in 'metacommunication' between human and computer (Veer *et al.*, 1985).

It is important to note that these styles and strategies should not only be considered as independent or intervening variables, but also as potentially dependent variables. Work can have a socializing effect (Frese, 1982), e.g. increasing planfulness, goal-orientation, reflectiveness, and algorithmic thinking.

Additionally, it has been suggested that work be organized differentially to fit the particular needs, styles, and concepts of the worker. This should not be thought of as a static concept in which some expert prescribes each person's adequate design, but as a dynamic process in which the individual develops and adapts the work situation (Ulich, 1983). Thus, this concept is related to the issue of control. With the advent of computer systems, an individualized path to work design seems to be more feasible than with traditional technology (Ulich, 1986). This implies that the system gives options. Aside from the default option (which may be particularly useful for the novice), the program should permit choices for how to do a task and which tools to use when. It should also be possible to 'reprogram' the system. Reprogramming does not mean that the source program is changed but that adaptations of the programs to one's individual needs are easily made. This implies, of course, that the user has learned the skills for 'reprogramming'. Moreover, individualization can imply that one can develop certain individualized paths of skill development at the workplace. While the latter suggestion may sound a bit utopian, there are indications in the literature that individualizing work increases work efficiency (Ackermann, 1986; Geiselman and Samet, 1982; Sasso, 1985; Zülch and Starringer, 1984).

GUIDELINES FOR HUMAN-COMPUTER INTERACTION AND THE DESIGN PROCESS

The Design Process

It has been repeatedly suggested that the interface design be radically separated from the design of the system *per se* (Branscomb and Thomas, 1984; Norman, 1983a). This makes it possible to change the interface design without having to change the whole system; thus, it can be improved after more knowledge on interface design has been accumulated. Furthermore, the design of the interface demands different skills from those required for the design of the system *per se* and might, therefore, be produced by different programmers.

However, this recommendation is not heeded very often. Moreover, most designers know very little about ergonomic considerations (Gould and Lewis, 1983) and use them even less. This is not just a 'bad' attitude towards the importance of psychology and human factors; there are also systematic reasons. Besides the obvious problems that there are technical constraints of limited memory and processing capacity, that designers work under very high time pressure, and that they usually are not really quite finished with the products in their own terms when the products are installed or arrive on the market (e.g. a product that still has bugs is thrown on the market), there are psychological constraints. The designers have difficulties in putting themselves into the shoes of the user, because the designers' tasks are different from those of the user (also the designer usually does not know much about the user's tasks; Smith and Mosier, 1984). Because of this, the qualifications and mental models of the designer are different as well.

The designer has a certain model in her mind of the system to be built. She then produces a system with a certain 'image'. It is this image that the user builds his model from (Norman, 1986a). The difficulty is that the user cannot directly perceive what the designer really meant, and problems could creep in while transforming the designer's model into a system and transforming the system's image into a user's model. (Hooper, 1986, actually uses the analogy of 'false façade' to characterize the image.) This issue becomes even more complicated when the psychologist enters the field and produces his own model of the system, of the user's model of the designer's model, etc. (Streitz, 1985). The designer and the user often do not both come up with that model of the system which is necessary to guarantee the functioning of the system. One important reason for this is that models are developed for certain tasks but the designer works on a different task from that of the user. This leads (again) to the demand that the user should be given high control over the system's functioning. In that case, the individual is able to adapt the system to his own model (and his own tasks), reducing the potential differences between the designer's model and his own (Greenberg and Witten, 1985; Huddleston, 1984; Raum, 1984; Rich, 1983; Ulich and Troy, 1986).

A variant of the differences between mental models is the fact that designers usually think about system-immanent problems (their tasks). The user on the other hand does not care about the system *per se* but only about its usefulness for doing his job. For example, the designer may emphasize a clean and logical structure of the program and the menus; but this 'cleanness' that the designer is proud of may actually impede the user's work because he has to plow through a lot of unnecessary menus (Hammond *et al.*, 1983). In addition to the designer being an expert in something different, the designer's expertise itself can have negative repercussions for the design process. Experts are usually bad teachers of novices because they have automized many procedures and are not able to verbalize them adequately, much less understand potential problems that a novice might have with them.

For all these reasons, psychology has to provide aids for the designers. There are essentially five approaches to help in the design of a program for human users; they are, of course, not mutually exclusive: (1) providing a set of quantitative psychological laws; (2) prototyping; (3) including the user in the design team; (4) providing theories, metaphors and analogies for the design; (5) giving guidelines and standards with a discussion of trade-offs. The last is probably the most important one and will therefore be considered separately and in a little more detail.

The first approach—providing a set of *quantitative laws*—has been suggested and followed by Card *et al.* (1983). This approach has several drawbacks from an industrial psychological point of view:

a. It is a question of whether their model can be adequately applied to the workplace. Just a single emotion (e.g. anger that the sequence of operations is prescribed in detail) would add an immeasurable amount of time to the postulated times for the cognitive processes. Similarly, a high-level goal, e.g. not wanting to work with a computer, has an impact on each low-level keystroke.
b. The low-level nature of their analysis and their little concern for ecological validity may lead to the wrong conclusions. Since they actually suppose that they have covered a large part of human-computer interaction, the lack of completeness of their approach is problematic. In their keystroke model, there is no systematic consideration of errors, 'nor are preferences for alternative command names, errors induced by complex command syntax, unusual sequencing of subtasks, comprehensibility of screen displays or menu structures, effectiveness of errors messages, help facilities, or documentation' considered (Shneiderman, 1984b, p. 236).
c. Since so many pieces for an applied science are missing, there is an aura of pseudo-exactness in their quantitative approach. Since quantitative approaches are preferred by computer scientists, this may lead to using those recommendations that are based on quantitative laws at the expense of other issues.
d. The emphasis is one-sided. The shortest possible command string (that follows from their model) may ignore issues of comprehensibility and memorability (Shneiderman, 1984b). Important (and high-level) design questions, such as why direct manipulation may be better than a simple command language approach, cannot be treated within their approach. It is also difficult to discuss trade-offs within their concepts (e.g. the trade-off between speed of work and creativity).
e. Productivity is seen primarily as a question of how many keystrokes or how many processors are involved. This implies a very short-term efficiency model which is typical of Taylor (1911) and Gilbreth (1919) and which does not take into account long-term needs. Thus, problems that are potentially associated with lower-level solutions may ensue; e.g. reducing the complexity of work may increase monotony (Greif and Holling, 1986).
f. Card *et al.* (1983) may have a wrong conception of the design process itself. Carroll and Rosson (1984) describe the design process as being neither in Card *et al.*'s sense top-down (using a general conception, e.g. a task description or a general law of psychology first and then delineating the steps from it) nor bottom-up (solving parts of the problem and then combining them into a whole), but as a mixture of all possible procedures. This often leads to a rejection of all solutions that had been entertained at the beginning of the design process.

Note that these points of critique do not call into question Card *et al.*'s substantive contributions in their specific areas. Obviously the level of analysis must depend on the question that is asked. If the research question is to design a keyboard then their model may be quite adequate. If the question is to integrate the computer in the workplace, their model is not so useful.

The (rapid) *prototyping* approach (e.g. Budde *et al.*, 1984; Gould and Lewis, 1983; Floyd, 1984; Richards, Boies, and Gould, 1986; Wixon *et al.*, 1983) is advantageous as it explicates potential user problems with the design at a very early stage of the game, thus enabling the designer to change the design and then test it again. Jörgensen (1984) argues for prototyping because designing a system is so complicated that one is never able to predict whether the system is satisfactory without empirically checking it. This presupposes, of course, that the relevant features are included in the prototype and that the relevant group of users is tested. Furthermore, a set of 'benchmark' tests or acceptance tests has to be developed (Carroll and Rosson, 1984; Roberts and Moran, 1983; Shneiderman, 1983b). Prototyping is an iterative process—a successive approximation to some goal (or benchmark). However, note that the iterative design process often includes changing one's goals (Carroll and Rosson, 1984). Although prototyping can be done without much theory behind it, it is useful to provide a background theory to analyze the results and draw the right conclusion from the evaluation of the prototype.

Potential pitfalls of this approach are that the prototypes are usually tested in an artificial context and that the prototype typically lacks certain necessary functions. Moreover, there is some debate whether users are really competent in providing ideas about good designs (Jörgensen, 1984).

Including the user in the design team is certainly one of the best ways to increase control for the user, to enhance mutual understanding between users and designers, and to ensure that the system is functional. Rapid prototyping is one form of inclusion, but including the user in the design team is another one. Potential drawbacks of this approach are that many users do not know the potentials of the computer, that the software is often designed for many different groups of users, and that the user may be too much concerned about her (novice) status now to suggest design ideas that will help her (as an expert) later.

Theories, metaphors, and analogies can aid the designer to use the right approaches—even if the 'nitty-gritty' of design is not explicated. Several analogies are discussed in Norman and Draper's (1986) book, e.g. architecture or theater. Two types of theories have been particularly useful for design consideration: keystroke theories (Card *et al.*, 1983; Hammer and Rouse, 1982; Schiele and Pelz, 1985) and action theories (Frese & Sabini, 1985; Hacker, 1985; Norman, 1986a; Rasmussen, 1983).

Guidelines

Guidelines and checklists are the most important aids for designers because they can orient the designer to the major problems and solutions of a user-centered design. These guidelines must be firmly rooted in psychological theory and research. There are now many published guidelines for improving different aspects of human-computer interaction (e.g. Branscomb and Thomas, 1984; Davis and Swezey, 1983; Döbele-Berger, Martin, and Martin, 1984; Dzida, 1985; Engel and Granada, 1975; Hannemyr and Innocent, 1985; Maguire, 1982; Spinas, Troy, and Ulich, 1983; Williges and Williges, 1983; Ulich, 1985). It is impossible to discuss them all or even a good number of them. Instead, I want, first, to suggest a hierarchy of guidelines and then to focus on one set of guidelines that have stirred up some controversies recently—the German suggestions for DIN Standards on the design of the human-computer dialogue.

A hierarchy of guidelines

The various guidelines discuss recommendations on quite different levels. It is, therefore, useful to provide some hierarchy among these guidelines as suggested in Table 5.2 (space constraints do not allow me to spell out the mesolevel and microlevel guidelines).

The assumption of Table 5.2 is that system design should be seen within the overall organizational framework. There should be participation in the introductory process. the organization should be decentralized so that more decisions can be made at each individual workplace; similarly, this applies to control over organizational decision making. Training needs to be adequate. Finally, it is a basic organizational decision which approach is taken *vis-à-vis* division of labor. From the standpoint of industrial psychology, the demand is put forward that the division of labor between the machine and human and between workers should be jointly optimized. This is, of course, an old demand of the sociotechnical approach (Emery and Thorsrud, 1976; Pava, 1983).

TABLE 5.2—The Hierarchy of Guidelines

Tier 1: Organizational Level
- —Participation in the introductory process
- —Decentralization
- —Control over organizational decision
- —Adequate training
- —Overall optimization of the human-human division and the human-computer division of labor

Tier 2: Workplace Level
- —Practicability
- —No damage to health or reduction of wellbeing
- —Providing for social interactions
- —Enhancement of personality

Tier 3: Task Level
- —Variety
- —Task significance
- —Task identity
- —Controllability over task decisions
- —Learning potential

Tier 4: Macrocriteria for Computer Systems (short-term/long-term)
- —Functionality
- —Usability
- —User friendliness

Tier 5: Mesocriteria for Computer Systems
- Example I: Specific criterion for usability: error reduction, error tolerance
- Example II: Controllability

Tier 6: Microcriteria for Computer Systems
- Example I: System understands synonyms for commands, provides an 'undo' command
- Example II: Default option plus easy change of layout, function keys, order of menus, menu content, reminders, command names, etc.

This means that division of labor should be reduced to allow higher variety, task significance and identity, controllability, and increase of learning potential. At each point, the question should be asked whether a technical decision has repercussions for the division of labor. This stands in contrast to just optimizing the technical subsystem—without regard to the workers involved.

Design should take into consideration the workplace as a whole. Therefore, the four criteria (suggested by different authors, e.g. Hacker, 1978; Ulich, 1986) of a good workplace are of importance. (By the way, we do not assume that the criteria suggested are orthogonal; they most probably correlate considerably and have a large overlap.) Practicability implies that the workplace is organized in such a way that the tasks can be accomplished (Hacker, 1978)—in a way, this is the workplace equivalent of the functionality of a computer system. The second criterion is that the workplace should not damage the workers' health (including psychosocial health) nor reduce their wellbeing. This issue is, of course, related to stress at work. A long-term stress at work (and low resources) may lead to ill-health and reduction of wellbeing. This is similarly true of lack of social interactions at the workplace. Finally, the workplace should allow one to advance one's personality (Hacker, 1978). Since the workplace has an influence on workers' intelligence and creativity, emotional growth and growth of self-confidence, and active (or passive) approaches to life (Frese, 1982), workplace design can be seen as enhancing (or thwarting) personality growth.

Four of the five criteria of the design of tasks are related to Hackman and Oldham's (1975) criteria for job motivation. Task variety signifies that different tasks (requiring different skills) are performed; task identity means that the worker completes a 'whole piece' of work rather than a meaningless part; task significance implies that the task is important for other people (or other people's work); controllability (Hackman and Oldham call it autonomy) refers to being able to decide on the content of the subtasks, on the order of the tasks, on the methods for solving the tasks, and on the timeframe in which to do the tasks. Finally, learning potential means that the tasks should be reasonably complex and should thus allow one to develop one's abilities and skills.

Roughly, the macrocriteria for the design of computer systems (Tier 4) can be grouped into three categories: Functionality, usability, and user friendliness. Each of them can have a short-term and a long-term meaning. (Of course, there is again an overlap between these different categories—often a functional system is easier to use and might have positive consequences to the user—but there are also differences.)

Functionality refers to whether a computer program allows and enhances the completion of the task. Thus, this term is oriented to the task outside the computer system. A short-term issue of functionality is, for example, whether the computer system models real-world tasks. A long-term issue is whether the user can redesign the system to fit the specifics of the tasks better (controllability) or whether the system can be adjusted to different approaches to the tasks. Some programs are high in functionality because they help to do the job, and low in usability, perhaps because of a bad command language (e.g. early spreadsheet programs; cf. Norman, 1986a).

Usability refers to whether the system is hard or easy to use. Examples of design issues of usability are tolerance of user errors, what kind of feedback the system gives, and whether it is consistent, self-explanatory, and corresponds to users'

expectations (see also Shackel, 1985b). In the short term, the issue of learnability is important. In the long run, adaptability to one's own style of working may be more important. It should be emphasized that the test of whether a system is usable has to be performed at the workplace, and not in the laboratory. For rather subtle reasons, the use of an apparently good system may be rejected (Eason, 1984).

Although this is not usually done, usability should be conceptually differentiated from *user friendliness* which means literally that the system is 'friendly' towards the user, i.e. that the system has no long- or short-term negative effects but positive effects on the user (it might also be liked best). A user-friendly system should not produce stress. The long-term effect is related to wellbeing and personality enhancement. For a short-term measure, human factor workers have frequently employed user satisfaction scales. However, user satisfaction is usually measured in a rather superficial manner. It would be productive to follow Bruggemann, Groskurth, and Ulich's (1975) suggestions on differentiating levels of satisfaction: (1) progressive satisfaction in which the level of aspiration of what the system should look like is increased; (2) stabilized satisfaction (level of aspiration is kept constant); (3) resigned satisfaction (level of aspiration is lowered to fit the system); (4) pseudo-satisfaction in which defense mechanisms prevail; (5) fixated dissatisfaction (dissatisfaction but no attempt to change the situation); and (6) constructive dissatisfaction in which the level of aspiration is kept up but one tries to change the system to match the aspirations.

We are not able to fully discuss Tiers 5 and 6 of the hierarchy because hundreds of recommendations apply at these levels. The examples may suffice. Error reduction is a more specific criterion of usability (and is related to functionality). Error reduction is achieved when synonyms for commands are recognized by the computer system and when there is an 'undo' command. Controllability may be related to functionality, usability, and user friendliness. Examples for controllability on the microlevel are given in Table 5.2.

The importance of presenting the guidelines as a hierarchical model is that each of these lower-level criteria can and should be related to the upper-level criteria. This underscores again the importance of organizational decisions.

The German guidelines on the human-computer dialogue

If the German suggestions for guidelines (DIN 66 234, part 8, 1984; see also Dzida, 1985, and Paetau, 1985) are approved officially, it will be one of the first attempts to nationally streamline the human factors consideration of software and encourage industry to respond. The standards are partly based on a study by Dzida, Herda, and Itzfeldt (1978). They have stirred up some controversies, since it is argued that it is too early to propose standards that might lead to inflexible use (e.g. Smith, 1986). The following five standards have been proposed—each one described by way of many examples: (1) task adequateness, which supports doing a task without adding load through system characteristics; (2) self-explanatory, i.e. the system is either immediately understandable or (full or partial) explanations are given on request; (3) user controllability, which implies that the user can modify the speed, and has a say in the number and the order of tools and the way the tasks are handled; (4) reliability, which means consistency with user expectations and internal consistency; and (5) error tolerance and transparency, i.e. the system accepts small errors and it explains when errors have occurred.

These standards clearly refer to different levels in the above hierarchy but they can be related to the criteria for a good workplace. Unlike some other German standards, these standards are relatively loose guidelines that are supposed to be optimized depending upon tasks and user groups. This stands in contrast to Smith's (1986) worries that they would lead to inflexibility. Obviously, there are trade-offs when applying these recommendations, as is true of all guidelines (Norman, 1983b). For example, a self-explanatory system may lack controllability since a system with maximal controllability can be changed into a new system that could not be predicted by the programmer (and thus no self-explanatory tools could be developed). Although these guidelines are supplemented by examples, designers may have difficulties using them, as is true of other guidelines (Mosier and Smith, 1986). It would be useful to develop data bases that could counsel the designers on each of their design tasks.

CONCLUSION

In this review the five major areas of interest to industrial and organizational psychology—organizational conditions, stress, cognitive optimization of human—computer interaction, individual differences, and design suggestions—have been summarized. The major conclusion has been that issues of human-computer interaction cannot and should not be separated from organizational issues. In the last analysis, organizational decisions on how the computer is used will contribute towards either positive or negative consequences in the employment of new technology. However, there is a sequel to this general statement: if the organization should, for example, decide on increasing control *vis-à-vis* a system, the question arises as to whether such a system exists and what should be the parameters of its design. Therefore, it is necessary that issues like control are taken seriously on each level—on the organizational level, the workplace level, the task level, and the levels of software ergonomics (macrocriteria, mesocriteria, and microcriteria for computer systems), and that concrete design suggestions are developed.

Unfortunately, as it turns out, organizational decisions are constrained less often by technology than by the marketing strategies of the hardware-producing companies (Cakir, 1981) and by restrictive conceptualizations of balancing costs and benefits. For example, when hardware was expensive, separate and centralized word-processing units were pushed by the producers and were estimated to be profitable. However, as pointed out by the economist Reichwald (1982), this 'profitability' turned out to be high only when the number of keystrokes was used as sole criterion of productivity. When other aspects were included—such as an increase in the total organizational time it takes to finish a letter, an increase in mistakes and in complicated and bureaucratic procedures, a reduction in the efficiency of specialists, etc.—centralized word-processing units are no longer seen to increase productivity. Developing national guidelines for software design, as in the Federal Republic of Germany, may enlarge our concept of cost-benefit analysis and encourage the production of adequate software that allows for different options and is functional, usable, and user friendly.

What can we say about the so-called 'big' controversies with which we started our review? From the perspective of industrial and organizational psychology, high-level (molar) approaches seem to be more adequate. However, low-level approaches can also be useful as long as they are subsumed under an overall organizational approach to the workplace.

The issue of control at work and controllability of the sytem is of high importance since control plays a role at the organizational level, the workplace and tasks levels, and each level of human-computer design. Furthermore, controllability has an impact not only on stress-effects, but also on performance and on the creativity with which an organization can accommodate to new environmental demands or to technological changes. Since it is likely that technological changes will be more frequent in the future, controllability for the individual workers has an ever more important function in the survival of industrial organizations.

The question of whether computers are seen as tools or as something beyond them may be dependent on the factor of controllability. If the computer is controllable and if the division of labor between human and machine is organized so that the human is firmly in control of the important procedures, decisions, and timeframes, it is more likely that the computer will be perceived as a tool. However, should controllability be reduced, then the computer is not a tool for those people who work with it, but only for the masters in the background. In such a case, stronger resistance introducing and using computers at the workplace is a more likely result.

This review has shown that we have already accumulated a fair amount of knowledge on some low-level issues of software design, but that these issues have to be integrated into a larger framework. Thus, it is necessary to develop an industrial and organizational psychology of computer use, system design, and integration of system design into an overall organizational design, so that we can minimize negative consequences of computer use and optimize productive and creative use of this powerful tool.

ACKNOWLEDGEMENTS

Thanks are due to L. Beerman, F. Glover, H. Gstalter, W. Kannheiser, J. Lang, and B. Zang for reading an earlier draft of the chapter and giving very useful suggestions.

REFERENCES

Ackermann, D. (1986). A pilot study on the effects of individualization in man-computer interaction. *Proceedings of the 2nd IFAC/IFIP/IFORS/IEA conference on analysis, design and evaluation. Varese.* London: Pergamon.

Anderson, J. R. (1983). *The Architecture of Cognition.* Cambridge, Massachusetts: Harvard University Press.

Baecker, R., Marcus, A. (1986). Design principles for the enhanced presentation of computer program source text. *Proceedings of the CHI '86 Conference on Human Factors in Computing Systems* (pp. 51-58). Boston.

Baitsch, C. (1986). Designing attempt 2: Personnel administration in private industry. In E. Ulich (ed.), *Computer-aided Office Work* (pp. 29-35). Zürich: ETH, Lehrstuhl für Arbeits- und Organisationspsychologie.

Balzert, H. (ed.) (1983). Software Ergonomie. *Berichte des German chapter of the ACM.* Stuttgart: Teubner.
Bannon, L. J. (1986). Helping users help another. In D. A. Norman and S. W. Draper (eds.), *User Centered Systems Design.* San Diego: Erlbaum.
Barnard, P. J., Hammond, N. V., MacLean, A., and Morton, J. (1982). Learning and remembering interactive commands in a text-editing task. *Behaviour and Information Technology,* **1,** 347-358.
Barnard, P. J., Hammond, N. V., Morton, J., and Long, J. B. (1981). Consistency and compatibility in human-computer dialogue. *International Journal of Man-Machine Studies,* **15,** 87-134.
Baroudi, J. J., Olson, M. H., and Ives, B. (1986). An empirical study of the impact of user involvement on system usage and information satisfaction. *Communications of the ACM,* **29,** 232-242.
Bartlett, F. C. (1932). *Remembering: A Study in Experimental and Social Psychology.* Melbourne: Cambridge University Press.
Bayman, P., and Mayer, R. E. (1984). Instructional manipulation of users' mental models for electronic calculators. *International Journal of Man-Machine Studies,* **20,** 189-199.
Benbasat, I., and Wand, Y. (1984). A structured approach to designing human-computer dialogues. *International Journal of Man-Machine Studies,* **21,** 105-126.
Bennett, J., Case, D., Sandelin, J., and Smith, M. (1984). *Visual Display Terminals: Usability Issues and Health Concerns.* Englewood Cliffs: Prentice Hall.
Bikson, T. K., and Gutek, B. A. (1983). Advanced office systems: An empirical look at use and satisfaction. *AFIPS Conference proceedings, National Computer Conference* (pp. 319-327).
Bjorn-Andersen, N. (1983). The changing roles of secretaries and clerks. In H. J. Otway and M. Peltu (eds.), *New Office Technology: Human and Organizational Aspects* (pp. 120-137). London: Frances Pinter.
Bjorn-Andersen, N., Eason, K., and Robey, D. (1986). *Managing Computer Impact: An International Study of Management and Organization.* Norwood, New Jersey: Ablex.
Bjorn-Andersen, N., Hedberg, B., Mercer, D., Mumford, E., and Sole, A. (1979). *The Impact of Systems Change in Organizations.* Alphen: Sijthoff and Noordhoff.
Bjorn-Andersen, N., and Rasmussen, L. B. (1980). Social implications of computer systems. In H. T. Smith and T. R. G. Green (eds.), *Human Interaction with Computers* (pp. 98-123). London: Academic Press.
Bjorn-Andersen, N. (1985). Training for subjection or participation. In B. Shackel (ed.), *Human-Computer Interaction* (pp. 839-846). Amsterdam: Elsevier.
Blauner, R. (1964). *Alienation and Freedom: The Factory Worker and His Industry.* Chicago: University of Chicago Press.
Boecker, H. D., Fischer, G., and Nieper, H. (1986). The enhancement of understanding through visual representations. *Proceedings of the CHI '86 Conference on Human Factors in Computing Systems* (pp. 44-50). Boston.
Boddy, D., and Buchanan, D. A. (1982). Information technology and the experience of work. In L. Bannon, U. Barry and O. Holst (eds.), *Information Technology* (pp. 144-157). Dublin: Tycooly International.
Boies, S., Wood, W., and Zimmer, W. (1986). Managing the design of user-computer interfaces. *Proceedings of the CHI '86 Conference on Human Factors in Computing Systems* (pp. 340-342). Boston.
Borman, L., and Curtis, B. (eds.) (1985). *Proceedings of the CHI '85 Conference on Human Factors in Computing Systems.* San Francisco.
Boucsein, W., Greif, S., and Wittekamp, J. (1984). Systemresponsezeiten als Belastungsfaktor bei Bildschirm-Dialogtätigkeiten. *Zeitschrift für Arbeitswissenschaft,* **38,** 113-122.
Boulay, du, B., and O'Shea, T. (1981). Teaching novices programming. In M. J. Coombs and J. L. Alty (eds.) *Computing Skills and User Interface* (pp. 147-201). NY: Academic Press.

Branscomb, L. M., and Thomas, J. C. (1984). Ease of use. A system challenge. *IBM Systems Journal,* **23,** 224-235.
Briefs, U., Ciborra, C., and Schneider, L. (eds.) (1983). *Systems Design for, with, and by the Users.* Amsterdam: North-Holland.
Brown, J. S. (1986). From cognitive to social ergonomics and beyond. In D. A. Norman and S. W. Draper (eds.), *User Centered Systems Design.* Hillsdale, Erlbaum.
Bruggemann, A., Groskurth. P., and Ulich, E. (1975). *Arbeitszufriedenheit.* Wien: Huber.
Buchanan, D. A., and Boddy, D. (1982). Advanced technology and the quality of working life: The effects of word processing on video typists. *Journal of Occupational Psychology,* **55,** 1-11.
Budde, R., Kuhlenkamp, K., Mathiassen, L., and Zülighoven, H. (1984). *Approaches to Prototyping.* Berlin: Springer.
Bullinger, H.-J. (ed.) (1985). *Software-Ergonomie '85.* Stuttgart: Teubner.
Cakir, A. (1981). Belastung und Beanspruchung bei Bildschirmtätigkeiten. In Frese, M. (ed.) *Stress im Büro* (pp. 46-71). Bern: Huber.
Cakir, A. (1986). Short paper towards an ergonomic design of software. *Behaviour and Information Technology,* **5,** 63-70.
Cakir, A. (ed.) (1983). *Bildschirmarbeit.* Berlin: Springer.
Cakir, A., Hart, D. J., and Stewart, T. F. M. (1979). *The VDT manual: Ergonomics, Workplace Design, Health and Safety, Task Organization.* Darmstadt: IFRA.
Card, S. K., Moran, T. P., and Newell, A. (1983). *The Psychology of Human-Computer Interaction.* Hilladale, NJ: Erlbaum.
Carroll, J. M. (1982). Learning, using and designing filenames and command paradigms. *Behaviour and Information Technology,* **1,** 327-346.
Carroll, J. M. (1984). *Mental Models and Software Human Factors: An Overview.* Yorktown NY: IBM Watson Research Center RC 10616.
Carroll, J. M. (1986). Science is soft at the frontier. *Talk at the ACM conference CHI '86.* Boston.
Carroll, J. M. and Carrithers, C. (1984). Training wheels in a user interface. *Communications of the ACM,* **27,** 800-806.
Carroll, J. M., and Mack, R. L. (1983). Actively learning to use a word processor. In W. E. Cooper (ed.), *Cognitive Aspects of Skilled Typewriting.* New York: Springer.
Carroll, J. M., and Mack, R. L. (1984). Learning to use a word processor: By doing, by thinking, and by knowing. In J. C. Thomas and M. L. Schneider (eds.), *Human Factors in Computer Systems* (pp. 13-51). Norwood, New Jersey: Ablex.
Carroll, J. M., Mack, R. L., Grischkowsky, N. L., and Robertson, S. R. (1985). Exploring exploring a word processor. *Human-Computer Interaction,* **1,** 283-307.
Carroll, J. M., and Mazur, S. A. (1985). *LisaLearning.* Yorktown: IBM Watson Research Center RC 11427.
Carroll, J. M., and Rosson, M. B. (1984). Usability specifications as a tool in development. In H. R. Hartson (ed.), *Advances in Human-Computer Interaction.* Norwood: Ablex Publishing.
Carroll, J. M., and Rosson, M. B. (in press). Paradox of the active user. In J. M. Carroll (ed.), *Interfacing Thought: Cognitive Aspects of Human-Computer Interaction.* Bradford Books/MIT Press.
Carroll, J. M., Smith-Kerker, P. L., Ford, J. R. and Mazur, S. A. (1986). *The Minimal Manual.* Yorktown: IBM Watson Research Center RC 11637.
Carroll, J. M., and Thomas, J. C. (1982). Metaphor and the cognitive representation of computing systems. *IEEE Transactions on Systems, Man, and Cybernetics,* **12,** 107-116.
Cashman, P. M. (1985). *Interfaces in organizations. Supporting Group Work.* San Francisco: Talk at the ACM conference CHI '85.
Chin, D. N. (1986). User modeling in UC, the UNIX consultant. *Proceedings of the CHI '86 Conference on Human Factors in Computing Systems.* Boston.

Coch, L., and French, J. R. P. (1948). Overcoming resistance to change. *Human Relations*, **19**, 39-56.
Coe, J. B., Cuttle, K., McClellon, W. C., Warden, N. J., and Turner, P. J. (1980). *Visual Display Units: A Review of Potential Problems Associated with their Use.* Wellington: New Zealand Department of Health.
Cohill, A. M., and Williges, R. C. (1985). Retrieval of HELP information for novice users of interactive computer systems. *Human Factors*, **27**, 335-343.
Cole, W. G. (1986). Medical cognitive graphics. *Proceedings of the CHI '86 Conference on Human Factors in Computing Systems* (pp. 91-95). Boston.
Coombs, M., and Alty, J. (1984). Expert systems: An alternative paradigm. *International Journal of Man-Machine Studies*, **20**, 21-43.
Cooper, C. L., and Marshall, J. (1976). Occupational sources of stress: A review of the literature relating to coronary heart disease and mental ill health. *Journal of Occupational Psychology*, **49**, 11-28.
Corbett, J. M. (1985). Prospective work design of human-centred CNC lathe. *Behaviour and Information Technology*, **4**, 201-214.
Cornelius, D. (1985). Scenario: Büro 1990. *Arbeitswissenschaft für Arbeitnehmer*. Düsseldorf: Deutscher Gewerkschaftsbund.
Dainhoff, M. J. (1982). Occupational stress factors in visual display terminal (VDT) operation: A review of empirical research. *Behaviour and Information Technology*, **1**, 141-176.
Dainoff, M. (1984). A model for human efficiency: Relating health, comfort and performance in the automated office workstation. In G. Salvendy (ed.), *Human Computer Interaction* (pp. 355-360). Amsterdam, Holland: Elsevier.
Dainoff, M. J., Happ, A., and Crane, P. (1981). Visual fatigue and occupational stress in VDT operators. *Human Factors*, **23**, 421-438.
Darlington, J., Dzida, W., and Herda, S. (1983). The role of excursions in interactive systems. *International Journal of Man-Machine Studies*. **18**, 101-112.
Davis, E. G., and Swezey, R. W. (1983). Human factors guidelines in computer graphics: A case study. *International Journal of Man-Machine Studies*, **18**, 113-133.
De Brabander, B., Vanlommel, E., Deschoolmeester, D., and Leyder, R. (1981). The impact of computer-use on organisation structure. In B. Shackel (ed.), *Man-Computer Interaction* (pp. 309-346). Dordrecht: Sigthoff and Noordhoff.
Dean, M. (1982). How a computer should talk to people. *IBM Systems Journal*, **21**, 424-453.
Dieckhoff, K., Dieckhoff, J., and Roth, V. (1982). Soziale Aspekte des technisch-organisatorischen Wandels in der Textverarbeitung. In Institut für angewandte Arbeitswissenschaft (ed.) *Folgewirkungen neuer Technologien in der Textverarbeitung*. Köln: Institut für angewandte Arbeitswissenschaft.
DIN 66 234 Teil 8 Entwurf (1984). *Bildschirmarbeitsplätze*: Grundsätze der Dialoggestaltung. (Deutsches Institut für Normung).
diSessa, A. A. (1986). Models of computation. In D. A. Norman and S. W. Draper (eds.), *User Centered Systems Design*. Hillsdale: Erlbaum.
Döbele-Berger, C., Martin, H., and Martin, P. (1984) *Gestaltung von Bildschirmmasken—Ableitung einer Checkliste*. Fachgebiet Arbeitswissenschaft, Gesamthochschule Kassel.
Douglas, S. A., and Moran, T. P. (1983). Learning text editor semantics by analogy. *Proceedings of the CHI '85 Conference on Human Factors in Computing Systems* (pp. 207-211). Boston.
Driscoll, J. W. (1982). How to humanize office automation. *Office: Technology and People*, **1**, 167-176.
Dumais, S. T., and Jones, W. P. (1985). A comparison of symbolic and spatial filing. *Proceedings of the CHI '85 Conference on Human Factors in Computing Systems* (pp. 127-130). San Francisco.
Dzida, W. (1985). Ergonomische Normen für die Dialoggestaltung. Wem nützen die Gestaltungsgrundsätze im Entwurf DIN 66234, Teil 8? In H.-J. Bullinger (ed.), *Software Ergonomie '85* (pp. 430-444). Stuttgart: Teubner.

Dzida, W., Herda, S., and Itzfeldt, W. D. (1978). User-perceived quality of interactive systems. *IEEE Transactions on Software Engineering SE-4*, 270-276.
Dzida, W., Hoffmann, C., and Valder, W. (1984). Wissensbasierte Dialogunterstützung. In H. Schauer and M. J. Tauber (eds.), *Psychologie der Computerbenutzung* (pp. 164-210). Wien: Oldenbourg.
Eason, K. D. (1982). The process of introducing information technology. *Behaviour and Information Technology*, **1**, 197-213.
Eason, K. D. (1984). Towards the experimental study of usability. *Behaviour and Information Technology*, **3**, 133-143.
Eller, E. C. (1984). Eine längsschnittanalytische Betrachtung von Einstellungen zur EDV und Arbeitszufriedenheit von Angestellten. In Berufsverband Deutscher Psychologen (ed.), *Arbeit in moderner Technik* (pp. 295-314). Duisburg: Eigendruck.
Ellis, P. (1984). Office planning and design: The impact of organizational change due to advanced information technology. *Behaviour and Information Technology*, **3**, 221-233.
Emery, F., and Thorsrud, E. (1976). *Democracy at Work: The Report of the Norwegian Industrial Democracy Program*. Leiden: Nijhoff.
Engel, S., and Granada, R. (1975). *Guidelines for Man/Display Interfaces*. New York: IBM Technical Report TR 00.2720.
Fähnrich, K.-P., and Ziegler, J. (1985). Direkte Manipulation als Interaktionsform an Arbeitsplatzrechnern. In H.-J. Bullinger (ed.), *Software-Ergonomie '85* (pp. 75-85). Stuttgart: Teubner.
Feigenbaum, E. A., and McCorduck (1984). *The Fifth Generation: Artificial Intelligence and Japan's Computer Challenge to the World*. New York: Signet.
Fischer, G., Lemke, A., and Schwab, T. (1985). Knowledge-based help systems. *Proceedings of the CHI '85 Conference on Human Factors in Computing Systems* (pp. 161-167). San Francisco.
Fitter, M., and Sime, M. (1980). Creating responsive computers: Responsibility and shared decision making. In H. T. Smith, and T. R. G. Green (eds.) *Human Interaction with Computers* (pp. 39-66). London: Academic Press.
Floyd, C. (1984). A systematic look at prototyping. In R. Budde, K. Kuhlenkamp, L. Mathiasen, and H. Züllighoven (eds.), *Approaches to Prototyping* (pp. 1-18). Berlin: Springer.
Foss, D. A., Rosson, M. B., and Smith, P. L. (1982). Reducing manual labor: Experimental analysis of learning aids for a text-editor. *Proceedings of the Conference on Human Factors of Computer Systems*. Gaithersberg, MD: National Bureau of Standarts.
Frese, M. (1982). Occupational socialization and psychological development: An underemphasized research perspective in industrial psychology. *Journal of Occupational Psychology*. **55**, 209-224.
Frese, M. (1984). Einstellungen zur Technologie und Computern: Entwicklung von Messinstrumenten. In Berufsverband Deutscher Psychologen (ed.), *Arbeit in moderner Technik* (pp. 97-108). Duisburg: Selbstverlag.
Frese, M. (1984). Transition in jobs, occupational socialization and stain. In V. L. Allen and E. v. d. Vliert (eds.), *Role Transitions* (pp. 239-252). New York: Plenum.
Frese, M. (in press) (a). A theory of control and complexity: Implications for software design and integration of computer systems into the work place. In M. Frese, E. Ulich and W. Dzida (eds.), *Psychological Issues of Human-Computer Interaction in the Work Place*. Amsterdam: North-Holland.
Frese, M. (in press) (b). *Stress at Work, Coping Strategies, and Musculoskeletal Complaints*.
Frese, M. (in prep.). *Social support as a moderator of the relationship between stress at work and psychological dysfuntioning.*

Frese, M., Albrecht, K., Altmann, A., Lang, J., v. Papstein, P., Peyerl, R., Prümper, J., Schulte-Göcking, H., Wankmüller, I., and Wendel, R. (1986). *The effects of an active development of the mental model in the training process: Experimental results on a word processing system*. München: Dept. of Psychology, University.

Frese, M., Saupe, R., and Semmer, N. (1981). Stress am Arbeitsplatz von Schreibkräften: Ein Vergleich zweier Stichproben. In M. Frese (ed.), *Stress im Büro* (pp. 225-251). Bern: Huber.

Frese, M., and Sabini, J. (eds.) (1985). *Goal Oriented Behavior: The Concept of Action in Psychology*. Hillsdale, NJ: Erlbaum.

Frese, M., Stewart, J., and Hannover, B. (1987). Goal-orientation and planfulness: Action styles as personality concepts. *Journal of Personality and Social Psychology*, **52**(6).

Frese, M., and Wendel, R. (in prep.). *Die Einführung von neuen Technologien am Arbeitsplatz: Gefahren und Chancen.*

Frese, M., and Zapf, D. (1987). Die Einführung von neuen Techniken am Arbeitsplatz verändert Qualifikationsanforderungen, Handlungsspielraum und Stressoren kaum. Ergebnisse einer Längsschnittuntersuchung. *Zeitschrift für Arbeitswissenschaften*, **41**, 7-14.

Gallistel, C. R. (1980). *The Organization of Action: A New Synthesis*. London: Lawrence Erlbaum.

Gallistel, C. R. (1985). Motivation, intention, and emotion: Goal directed behavior from a cognitive-neuroethological perspective. In M. Frese and J. Sabini (eds.), *Goal Directed Behavior* (pp. 48-66). Hillsdale: Erlbaum.

Geiselman, R. E., and Samet, M. G. (1982). Personalized versus fixed formats for computer-displayed intelligence messages. *IEEE Transactions on Systems, Man, and Cybernetics, SMC-12*, 490-495.

Gilbreth, F. B. (1919). *Applied motion study*. New York: Macmillan.

Giuliano, V. E. (1982). The mechanization of office work. *Scientific American*, **247**, 148-165.

Glenn, E. N., and Feldberg, R. L. (1977). Degraded and deskilled: The proletarization of clerical work. *Social Problems*, **25**, 52-64.

Gomez, L. M., and Egan, D. E. (1983). How interface design determines who has difficulty learning to use a text editor. (pp. 176-181) *Proceedings of the CHI '83 Conference on Human Factors in Computing Systems*. Boston.

Gorny, P. (1984). Zur Manipulation visueller Information. In H. Schauer and M. J. Tauber (eds.), *Psychologie der Computerbenutzung* (pp. 55-88). Wien: Oldenbourg.

Gottschall, K., Mickler, O., and Neubert, J. (1985). *Computerunterstützte Verwaltung*. Frankfurt: Campus.

Gould, J. D., and Lewis, C. (1983). Designing for usability-key principles and what designers think. *Proceedings of the CHI '85 Conference on Human Factors in Computing Systems* (pp. 50-53). Boston.

Gould, J. D., Lewis, C., and Barnes, V. (1985). Effects of cursor speed on text-editing. *Proceedings of the CHI '85 Conference on Human Factors in Computing Systems*. (pp. 7-10). San Francisco.

Green, T. R. G., Payne, S. J., and v.d. Veer, G. C. (eds.) (1983). *The Psychology of Computer Use*. London: Academic Press.

Greenberg, S., and Witten, I. H. (1985). Adaptive personalized interface—A question of viability. *Behaviour and Information Technology*, **4**, 31-45.

Gregory, J., and Nussbaum, K. (1982). Race against time: Automation of the office. *Office: Technology and People*, **1**, 197-236.

Greif, S. (1986a). Job design and computer training. *Bulletin of the British Psychological Society*, **39**, 166-169.

Greif, S. (1986b). Neue Kommunikationstechnologien—Entlastung oder mehr Stress? In K. Pullig, U. Schaetzel, and J. Scholz (eds.), *Stress, Reihe Betriebswirtschaftliche Weiterbildung* (Vol. 8). Hamburg: Windmuehle Verlag.

Greif, S., and Holling, H. (1986). Neue Technologien. In D. Frey and S. Greif (eds.), *Sozialpsychologie* (2nd ed.) München: Urban and Schwarzenberg.

Grudin, J., and Banard, P. (1985). When does an abbreviation become a word? And related questions. *Proceedings of the CHI '85 Conference on Human Factors in Computing Systems* (pp. 121-125). San Francisco.

Gunnarson, E. (1984). The impact of organisational factors on visual strain in clerical VDT work. In B. G. F. Cohen (ed.), *Human Aspects in Office Automation* (pp. 43-61). Amsterdam: Elsevier.

Hacker, W. (1978). *Allgemeine Arbeits- und Ingeneurpsychologie* (2nd ed.). Bern: Verlag Hans Huber.

Hacker, W. (1983). Psychische Beanspruchungen bei Text- und Datenverarbeitungstätigkeiten an Bildschirmgeräten: Ermittlung und Gestaltung. *Zeitschrift für Psychologie Supplement*, **5**, 24-41.

Hacker, W. (1985). Activity: A fruitful concept in industrial psychology. In M. Frese and J. Sabini (eds.), *Goal Directed Behavior* (pp. 262-284). Hillsdale: Erlbaum.

Hacker, W., Rühle, R., and Schneider, N. (1978). Effektivitätssteigerung durch neue Verfahren zum Erlernen leistungsbestimmender geistiger Teiltätigkeiten. *Sozialistische Arbeitswissenschaft*, **22**, 363-368.

Hackman, J. R., and Oldham, G. R. (1975). Development of job diagnostic survey. *Journal of Applied Psychology*. **60**, 159-170.

Haider, E., and Rohmert, W. (1983). Beziehungen zwischen Technologien und Organisationsmerkmalen in der Textverarbeitung. *Zeitschrift für Arbeitswissenschaft*, **37**, 211-214.

Haider, M., Kundi, M., and Weissenböck, X. (1980). Work related strain to VDU with different colored characters. In E. Grandjean and E. Vigliani (eds.), *Ergonomic Aspects of Visual Display Terminals*. London: Taylor and Francis.

Hammer, J. M., and Rouse, W. B. (1982). The human as a constrained optimal editor. *IEEE Transactions on Systems, Man, and Cybernetics, SMC-12*, 777-784.

Hammond, N., and Barnard, P. (1984). Dialogue design: Characteristics of user knowledge. In A. Monk (ed.), *Fundamentals of Human-Computer Interaction* (pp. 127-164). London: Academic Press.

Hammond, N., Jorgenson, A., MacLean, A., Barnard, P., and Long, J. (1983). Design practice and interface usability: Evidence from interviews with designers. *Proceedings of the CHI '83 Conference on Human Factors in Computing Systems* (pp. 40-44). Boston.

Hannemyr, G. and Innocent, P. R. (1985). A network user interface: Incorporating human factors guidelines into the ISO standard for open systems interconnection. *Behaviour and Information Technology*, **4**, 309-326.

Hartmann, A. L., and Zwahlen, H. T. (1985). Mehr Augenbeschwerden durch Bildschirmarbeit? *Sozial- und Präventivmedizin*, **30**, 280-281.

Hauptmann, A. G., and Green, B. F. (1983). A comparison of command, menu-selection and natural-language computer programs. *Behaviour and Information Technology*, **2**, 163-178.

Hayes, P. J., and Reddy, R. D. (1983). Steps toward graceful interaction in spoken and written man-machine communication. *International Journal of Man-Machine Studies*, **19**, 231-284.

Hayes-Roth, R., Waterman, D. A., and Lenat, D. B. (eds.) (1983). *Building Expert Systems*. London: Addison-Wesley.

Hedberg, B. (1980). Using computerized information systems to design better organizations and jobs. In Bjorn-Andersen, N. (ed.), *The Human Side of Information Processing* (pp. 19-37). Amsterdam: North-Holland.

Helmreich, R. (1985). Human aspects of office systems: User acceptance research results. In B. Shackel (ed.), *Human-Computer Interaction* (pp. 715-718). Amsterdam: Elsevier.

Hiltz, S. R. (1983). A study of the determinants of acceptance of computer-medited communication systems. *SIGCHI Bulletin*, **14**, 16-17.

Hirschheim, R. A., Land, F. F., and Smithson, S. (1985). Implementing computer-based information systems in organizations: Issues and strategies: In B. Shackel (ed.), *Human-Computer Interaction* (pp. 855-863) North-Holland.

Hooper, K. (1986). Architectural design: An analogy. In D. A. Norman and S. W. Draper (ed.), *User-Centered System Design*. Hillsdale: L. Erlbaum.

Hoos, I. R. (1960). When the computer takes over the office. *Harvard Business Review*, **38**, 102-112.

Hoos, I. R. (1983). When the computer takes over the office—Update. *Office: Technology and People*, **2**, 69-77.

Howath, P. A., and Istance, H. O. (1985). The association between visual discomfort and the use of visual display units. *Behaviour and Information Technology*, **4**, 131-149.

House, J. S. (1981). *Work Stress and Social Support*. London: Addison-Wesley.

Huddleston, J. H. F. (1984). What individuality means for systems design. *Behaviour and Information Technology*, **3**, 85-91.

Hutchins, E., Hollan, J. D., and Norman, D. A. (1986). Direct manipulation interfaces. In D. A. Norman and S. W. Draper (eds.), *User-Centered System Design*. Hillsdale: Erlbaum.

Iacono, S., and Kling, R. (in press). Changing office technologies and transformation of clerical jobs: A historical perspective. In R. Kraut (ed.), *Technology and the Transformation of White-Collar Work*. Hillsdale: Erlbaum.

Isa, B. S., Boyle, J. M., Neal, A. S., and Simons, R. M. A methodology for objectively evaluating error messages. *Proceedings of the CHI '83 Conference on Human Factors in Computing Systems* (pp. 68-71). Boston.

Jagodzinski, A. P. (1983). A theoretical basis for the representation of on-line computer systems to naive users. *International Journal of Man-Machine Studies*, **18**, 215-252.

Janda, A. (1983). *Proceedings of the CHI '83 Conference on Human Factors in Computing Systems*. Boston.

Jörgensen, A. H. (1984). On the psychology of prototyping. In Budde, R., Kuhlenkamp, K., Mathiassen, L., and Züllighoven, H. (eds.), *Approaches to Prototyping* (pp. 278-289). Berlin: Springer.

Johansen, R., and Baker, E. (1984). User needs workshops: A new approach to anticipating user needs for advanced office systems. *Office: Technology and People*, **2**, 103-119.

Johansson, G., and Aronsson, G. (1984). Stress reactions in computerized administrative work. *Journal of Occupational Behaviour*. **5**, 159-181.

Jones, W. P., and Landauer, T. K. (1985). Context and self-selection effects in name learning. *Behaviour and Information Technology*, **4**, 3-17.

Kalimo, R., and Leppänen, A. (1985). Feedback from VDTs, performance control and stress in text preparation in the printing industry. *Journal of Occupational Psychology*, **58**, 27-38.

Karasek, R. A. (1979). Job demands, job decision latitude and mental strain: Implications for job redesign. *Administrative Science Quarterly*. **24**, 285-308.

Karasek, R. A., Bader, D., Marxner, R., Ahlbom, A., and Theorell, T. (1981). Job design latitude, job demands, and cardiovascular disease: A prospective study of Swedish men. *American Journal of Public Health*, **71**, 634-705.

Karat, J., Boyes, L., Weisgerber, S., and Schafer, C. (1986). Transfer between word processing systems. *Proceedings of the CHI '86 Conference on Human Factors in Computing Systems* (p. 67-71). Boston.

Katz, C. (1986). Office of the future: Trends and premises. In E. Ulich (ed.), *Computer-aided Office Work* (pp. 36-39). Zürich. ETH, Lehrstuhl für Arbeits- und Organisatiospsychologie.

Kaye, A. R., and Sutton, M. J. D. (1985). Productivity and quality of working life for office principals and the implications for office automation. *Office: Technology and People*, **2**, 267-286.

Kempton, W. (1986). Two theories of home heat control. *Cognitive Science*, **10**, 75-90.

Kern, H., and Schumann, M. (1984). *Das Ende der Arbeitsteilung? Rationalisierung in der industriellen Produktion: Bestandsaufnahme, Trendbestimmung*. München: Beck.

Kieras, D. E., and Bovair, S. (1984). The role of a mental model in learning to operate a device. *Cognitive Science*, **8**, 255-273.

Kiger, J. I. (1984). The depth/breadth trade-off in the design of menu-driven user interfaces. *International Journal of Man-Machine Studies*, **20**, 201-213.

Kling, R. (1980). Social analyses of computing: Theoretical perspectives in recent empirical research. *Computing Surveys*, **12**, 61-110.

Klockare, B., and Norrby, K. (1983). A swedish model for systems development in public administration. In U. Briefs, C. Ciboorra and L. Schneider (eds.), *Systems Design for, with, and by the Users* (pp. 119-127). Amsterdam: North-Holland Publishing Company.

Koved, L., and Shneiderman, B. (1986). Embedded menus: Selecting items in context. *Communications of the ACM*, **29**, 312-318.

Kuhmann, W., Boucsein, W., Schäfer, F., and Alexander, J. (in press). Experimental investigation of psychophysiological stress reactions induced by different system-response times in man-computer-interaction. *Ergonomics*.

Läubli, T., Hünting, W., and Grandjean, E. (1980). Visual impairments in VDU operators related to environmental conditions. *Ergonomics*, **24**, 933.

Landauer, T. K., and Nachbar, D. W. (1985). Selection from alphabetic and numeric menu trees using a touch screen: Breath, depth and width. *Proceedings of the CHI '85 Conference on Human Factors in Computing Systems* (pp. 73-78). San Francisco.

Landy, F. L. (1985). *Psychology of Work Behavior*. Homewood, Illinois: The Drosey Press.

Lang, K., Auld, R., and Lang, T. (1982). The goals and methods of computer users. *International Journal of Man-Machine Studies*, **17**, 375-399.

Lange, K. (1984). Zwischen Hoffen und Bangen. *Bild der Wissenschaft*. (1), 63-71.

Lewis, C., and Norman, D. A. (1986). Designing for error. In D. A. Norman and S. W. Draper (eds.), *User-Centered System Design*. Hillsdale: Erlbaum.

Long, J., Hammond, N., Barnard, P., Morton J., and Clark, I. (1983). Introducing the interactive computer at work. *Behaviour and Information Technology*, **2**, 39-106.

Maass, S. (1983). Why systems transparency? In T. R. G. Green, S. J. Payne and G. C. van der Veer (eds.), *The Psychology of Computer Use* (pp. 19-28). London: Academic Press.

Mack, R. (1984). *Understanding Text-Editing: Evidence from Predictions and Description Given by Computer-Naive People*. Yorktown: IBM Watson Research Center RC 10333.

Mack, R. (1985). Identifying and designing toward new user expectations in a prototype text-editor. *Proceedings of the CHI '85 Conference on Human Factors in Computing Systems* (pp. 139-141). San Francisco.

Mack, R. L. Lewis, C. H., and Carroll, J. M. (1983). Learning to use word processors: Problems and prospects. *ACM Transactions on Office Information Systems*, **1**, 254-271.

Maguire, M. (1982). An evaluation of published recommendations on the design of man-Computer dialogues. *International Journal of Man-Machine Studies*, **16**, 237-261.

Mantei, M., and Orbeton, Peter (eds.) (1986) *Proceedings of the CHI '86 Conference on Human Factors in Computing Systems*. Boston.

Margulies, F., and Zemanek, H. (1983). Man's role in man-machine systems. In G. Johannsen and J. E. Rijnsdorp (eds.) *IFAC Analysis, Design, and Evaluations of Man-Machine Systems* (pp. 121-128). Oxford, Pergamon.

Martin, G. L., and Corl, K. G. (1986). System response time effects on user productivity. *Behaviour and Information Technology*, **5**, 3-13.

Mayer, R. E. (1975). Different problem-solving competencies established in learning computer programming with and without meaningful models. *Journal of Educational Psychology*, **67**, 725-734.
Mayer, R. E. (1976). Some conditions of meaningful learning for computer programming: Advanced organizers and subject control of frame order. *Journal of Educational Psychology*, **68**, 143-150.
McKendree, J., and Carroll, J. M. (1986). Advising roles of computer consultant. *Proceedings of the CHI '86 Conference on Human Factors in Computing Systems* (pp. 35-40). Boston.
Miller, G. A. (1956). The magical number seven, plus or minus two: Some limits on our capacity for processing information. *Psychological Review*, **63**, 81-97.
Miller, G. A., Galanter, E., and Pribram, K. H. (1960). *Plans and the Structure of Behavior*. London: Holt.
Miller, R. B. (1968). Response time in man-computer conversational interactions. *Proceedings Spring Joint Computer Conference*, **33**, (pp. 267-277). Montvale: AFIPS Press.
Miyake, N., and Norman, D. A. (1979). To ask a question, one must know enough to know what is not known. *Journal of Verbal Learning and Verbal Behavior*, **18**, 357-364.'
Monk, A. (ed.) (1984). *Fundamentals of Human-Computer Interaction*. London: Academic Press.
Moran, T. P. (1981). An applied psychology of the user. *Computing Surveys*, **13**, 1-11.
Morland, V. D. (1983). Human factors guidelines for terminal interface design. *Communications of the ACM*, **26**, 484-494.
Mosier, J. N., and Smith, S. L. (1986). Application of guidelines for designing user interface software. *Behaviour and Information Technology*, **5**, 39-46.
Mowshowitz, A. (1976). *The Conquest of Will, Information Processing in Human Affairs*. Reading: Addison-Wesley.
Müller-Böling, D. (1984). Durch bessere Technik zu mehr Akzeptanz. *Office Management*, **11**, 1064-1066.
Mumford, E. (1980). The participative design of clerical information systems. In N. Bjorn-Andersen (ed.), *The Human Side of Information Processing* (pp. 91-107). Amsterdam: North-Holland.
Neisser, U. (1976). *Cognition and Reality: Principles and Implications of Cognitive Psychology*. San Francisco: W. H. Freeman.
Neisser, U. (1985). The role of invariant structures in the control of movement. In M. Frese and J. Sabini (eds.), *Goal directed behavior* (pp. 97-109). Hillsdale: Erlbaum.
Newell, A., and Card, S. K. (1985). The prospects for psychological science in human-computer interaction. *Human-Computer Interaction*, **1**, 209-242.
Nickerson, R. S. (1986). *Using Computers: The Human Factors of Information Systems*. London: The MIT Press.
Norman, D. A. (1981). Categorization of action slips. *Psychological Review*, **88**, 1-15.
Norman, D. A. (1983a). Some observations on mental models. In D. Gentner and A. L. Stevens (eds.), *Mental Models* (pp. 7-14). Hillsdale: Erlbaum.
Norman, D. A. (1983b). Design principles for human-computer interfaces. *Proceedings of the CHI '83 Conference on Human Factors in Computing Systems* (pp. 1-10). Boston.
Norman, D. A. (1984a). Stages and levels in human-machine interaction. *International Journal of Man-Machine Studies*, **21**, 365-375.
Norman, D. A. (1984b). *Working Papers on Errors and Error Detection*. San Diego: Univ. of California.
Norman, D. A. (1986a). Cognitive engineering. In D. A. Norman and S. W. Draper (eds.), *User Centered Systems Designs*. Hillsdale: Erlbaum.
Norman, D. A. (1986b). Interface design doesn't matter. *Talk at the ACM conference CHI '86*. Boston.
Norman, D. A., and Draper, S. W. (eds.) (1986). *User Centered System Design*. Hillsdale: Erlbaum.

Nullmeier, E., and Rödiger, K.-H. (1985). *Bibliographie zur Gestaltung von Dialogschnittstellen*. Berlin: Technische Universität Berlin.
O'Malley, C., Smolensky, P., Bannon, L., Conway, E., Graham, J., Sokolov, J., and Monty, M. L. (1983). A proposal for user centered system documentation. *Proceedings of the CHI '83 Conference on Human Factors in Computing Systems*. Boston.
O'Malley, C. (1986). Helping users help themselves. In D. A. Norman and S. W. Draper (eds.), *User Centered Systems Design*. Hillsdale: Erlbaum.
Otway, H. J., and Peltu, M. (1983). *New Office Technology: Human and Organizational Aspects*. London: Pinter.
Owen, D. (1986). Naive theories of computation. In D. A. Norman and S. W. Draper (eds.), *User Centered System Design*. Hillsdale: Erlbaum.
Paetau, M. (1985). The cognitive regulation of human action as a guideline for evaluating the man-compute dialogue. In B. Shackel (ed.), *Human-Computer Interaction* (pp. 731-736). Amsterdam: North-Holland.
Pask, G., and Scott, B. C. E. (1972). Learning strategies and individual competence. *International Journal of Man-Machine Studies*, **4**, 217-253.
Pava, C. H. P. (1983). *Managing New Office Technology*. New York: Free Press.
Pelzmann, L., Winkler, N., and Zewell, E. (1985). Antizipation von Arbeitslosigkeit. In T. Kieselbach and A. Wacker (eds.), *Individuelle und gesellschaftliche Kosten der Massenarbeitslosigkeit* (pp. 256-268). Weinheim: Beltz.
Picot, A., Klingenberg, H., and Kränzle, H. (1982). Office technology: A report on attitudes and channel selection from field studies in Germany. In M. Burgoon (ed.), *Communication Yearbook* Vol. 6 (pp. 674-692). London: Sage Publications.
Pollak, M. E. (1985). Information sought and information provided: An empirical study of user/expert dialogues. *Proceedings of the CHI '85 Conference on Human Factors in Computing Systems* (pp. 155-159). San Francisco.
Polson, P. G., and Kieras, D. E. (1985). A quantitive model of the learning and performance of text editing knowledge. *Proceedings of the CHI '85 Conference on Human Factors in Computing Systems* (pp. 207-212). San Francisco.
Pomfrett, S. M., Olphert, C. W., and Eason, K. D. (1985). Work organisation implications of word processing. In B. Shackel (ed.), *Human-Computer Interaction* (pp. 847-854). Amsterdam: Elsevier.
Powers, M., Lashley, C., Sanchez, P., and Schneiderman, B. (1984). An experimental comparison of tabular and graphic data presentation. *International Journal of Man-Machine Studies*, **20**, 545-566.
Preece, J. (1983). Graphs are not straightforward. In T. R. G. Green, S. J. Payne and G. C. v.d. Veer (eds.), *The Psychology of Computer Use* (pp. 41-56). London: Academic Press.
Price, H. E. (1985). The allocation of functions in systems. *Human Factors*, **27**, 33-45.
Radl G. W. (undated). *Zur Notwendigkeit der Akzeptanzforschung bei der Entwicklung neuer Systeme der Bürotechnik*. Düsseldorf: Akzente.
Rafaeli, A., and Sutton, R. I. (1986). Word processing technology and perceptions of control among clerical workers. *Behaviour and Information Technology*, **5**, 31-37.
Ranney, J. M., and Carder, C. E. (1984). Socio-technical design methods in office settings: Two cases. *Office: Technology and People*, **2**, 169-186.
Rasmussen, J. (1983). Skills, rules and knowledge; signals, signs, and symbols, and other distinctions in human performance models. *IEEE Transactions on Systems, Man, and Cybernetics, SMC-13*, 257-266.
Raum, H. (1984). Aufgabenabhängige Gestaltung des Informationsangebots bei Bildschirmarbeit. *Schweizerische Zeitshrift für Psychologie und ihre Anwendungen*, **43**, 25-33.
Reichwald, R. (undated). *Zur Notwendigkeit der Akzeptanzforschung bei der Entwicklung neuer Systeme der Bürotechnik*. Düsseldorf: Akzente.

Reichwald, R. (1982). Neue Systeme der Bürotechnik und Büroarbeitsgestaltung: Problemzusammenhänge. In R. Reichwald (ed.), *Neue Systeme der Bürotechnik*. Berlin: Schmidt.
Reichwald, R. (1983). Bürotechnik, Bürokratisierung und das Zentralisierungsproblem: Grundüberlegungen zur Gestaltung der Büroarbeit. In A. Cakir (ed.), *Bildschirmarbeit* (pp. 23-46). Berlin: Springer.
Rich, E. (1983). Users are individuals: Individualizing user models. *International Journal of Man-Machine Studies*, **18**, 199-214.
Richards, J. T., Boies, S. J., and Gould, J. D. (1986). Rapid prototyping and systems development: Examination of an interface toolkit for voice and telephone applications. *Proceedings of the CHI '86 Conference on Human Factors in Computing Systems* (pp. 216-220). Boston.
Riley, M. (1986). User understanding. In D. A. Norman and S. W. Draper (eds.), *User Centered Systems Design*. Hillsdale: Erlbaum.
Roberts, T. L., and Moran, T. P. (1983). The evaluation of text editors: Methodology and empirical results. *Communications of the ACM*, **26**, 265-283.
Robertson, I. T. (1985). Human information-processing strategies and style. *Behaviour and Information Technology*. **4**, 19-29.
Rödiger, K. H. (1985). Beiträge der Softwareergonomie zu den frühen Phasen der Software-Entwicklung. In H.-J. Bullinger (ed.), *Software-Ergonomie '85* (pp. 455-464). Stuttgart: Teubner.
Rohr, G. (1984). Understanding visual symbols. *Proceedings of IEEE workshop on visual languages* (pp. 184-191). Hiroshima.
Rohr, G., and Tauber, M. J. (1984). Representational frameworks and models for human-computer interfaces. In G. C. van der Veer, M. J. Tauber, T. R. G. Green, and P. Gorny (eds.), *Readings on Cognitive Ergonomics: Mind and Computers* (pp. 8-26). Berlin: Springer.
Rosenberg, J. K. (1982). Evaluating the suggestiveness of command names. *Behaviour and Information Technology*, **1**, 371-400.
Rosenberg, J. K., and Moran, T. P. (1985). Generic commands. In B. Shackel (ed.), *Human--Computer Interaction* (pp. 254-250). Amsterdam: Elsevier.
Rosenstiel, v. L. (1984). Aufgaben der Arbeits- und Betriebspsychologie bei sich wandelnden Technologien, Organisationstrukturen und Werthaltungen. In Berufsverband Deutscher Psychologen (ed.), *Arbeit in moderner Technik* (pp. 15-50). Duisburg: Eigendruck.
Ruch, L. (1986). The use of word processors: A survey in Switzerland's Federal Government Administration. In E. Ulich (ed.), *Computer-aided Office Work* (pp. 17-23). Zürich: ETH, Lehrstuhl für Arbeits- und Organisationspsychologie.
Sabini, J., and Silver, M. (1985). On the captivity of the will: Sympathy, caring, a moral sense of the human. *Journal for the Theory of Social Behaviour*, **15**, 24-37.
Salvendy, G. (1984). Research issues in the ergonomics, behavioral, organizational and management aspects of office automation. In B. G. F. Cohen (ed.), *Human Aspects in Office Automation* (pp. 115-127). Amsterdam: Elsevier.
Sasso, R. (1985). Personalising the software interface. In B. Shackel (ed.), *Human-Computer Interaction* (pp. 355-362). Amsterdam: Elsevier.
Sauter, S., Gottlieb, M. S., Jones, K. C., Dodsen, V., and Rohrer, K. M. (1983). Job and health implications of VDT use: Initial results of the Wisconsin-NIOSH Study. *Communications of the ACM*, **26**, 284-294.
Scapin, D. L. (1981). Computer commands in restricted natural language: Some aspects of memory and experience. *Human Factors*, **23**, 365-375.
Scapin, D. L. (1982). Generation effect, structuring, and computer commands. *Behaviour and Information Technology*, **1**, 401-410.
Schaefer, F., Kuhmann, W., Boucsein, W., and Alexander, J. (1986). Beanspruchung durch Bildschirmtätigkeit bei experimentell variierten Systemresponsezeiten. *Zeitschrift für Arbeitswissenschaft*, **40**, 31-38.

Schardt, L. P., and Knepel, W. (1981). Psychische Beanspruchungen kaufmännischer Angestellter bei computergestützter Sachbearbeitung. In M. Frese (ed.), *Stress im Büro* (pp. 125-158). Bern: Huber.
Scharer, L. L. (1983). User training: Less is more. *Datamation*, **29**, 175-182.
Schauer, H., and Tauber, M. (eds.) (1983). *Psychologie des Programmierens*. Wien: Oldenbourg.
Schauer, H., and Tauber, M. (eds.) (1984). *Psychologie der Computerbenutzung*. Wien: Oldenbourg.
Schiele, F., and Pelz, W. H. (1985). Eine Studie zue empirischen Überprüfung der Benutzerfreundlichkeit von Textverarbeitungs- und Tabellenkalkulationsprogrammen. In H.-J. Bullinger (ed.), *Software Ergonomie '85* (pp. 239-250). Stuttgart: Teubner.
Schönpflug, W. (1985). Goal directed behavior as a source of stress: Psychological origins and consequences of inefficiency. In M. Frese and J. Sabini (eds.), *Goal Directed Behavior* (pp. 172-188). Hillsdale: Erlbaum.
Schulte-Göcking (1987). *Lernprozesse an Textsystemen: Der Einfluss von Lernstil, Handlungsstil, Problemlösekompetenz und Persönlichkeitsmerkmalen auf den Lernerfolg*. München: Institut für Psychologie, Univ. München.
Seligman, M. B. P. (1975). *Learning helplessness*. San Francisco: Freeman.
Semmer, N. (1982). Stress at work, stress in private life and psychological well-being. In W. Bachmann and I. Udris (eds.), *Mental Load and Stress in Activity: European Approaches* (pp. 42-55). Amsterdam: Elsevier.
Semmer, N. (1984). Stressbezogene Tätigkeitsanalyse: Psychologische Untersuchungen zur Analyse von Stress am Arbeitsplatz. Weinheim: Beltz.
Semmer, N., and Frese, M. (1985). Action theory in clinical psychology. In M. Frese and J. Sabini (eds.), *Goal Directed Behavior* (pp. 296-310). Hillsdale: Erlbaum.
Shackel, B. (1985a). *Human-Computer Interaction*. Amsterdam: Elsevier, North-Holland.
Shackel, B. (1985b). Human factors and usability: Whence and whither. In H.-J. Bullinger (ed.), *Software-Ergonomie '85* (pp. 1-13). Stuttgart: Teubner.
Shneiderman, B. (1980). *Software Psychology*, Cambridge, Massachusetts: Winthrop Publishers.
Shneiderman, B. (1982a). System message design: Guidelines and experimental results. In H. Badre and B. Shneiderman (eds.), *Directions in Human-Computer Interaction* (pp. 55-78). Norwood New Jersey. Ablex Publishing Co.
Shneiderman, B. (1982b). The future of interactive systems and the emergence of direct manipulation. *Behaviour and Information Technology*, **1**, 237-256.
Shneiderman, B. (1983a). Direct manipulation: A step beyond programming languages. *Computer*, **16**, 57-69.
Shneiderman, B. (1983b). Human factors of interactive software. In G. Goos and J. Hartmanis (eds.), *Lecture Notes in Computer Science* (pp. 9-29). Berlin: Springer.
Shneiderman, B. (1984a). Response time and display rate in human performance with computers. *Computing Survey*, **16**, 265-285.
Shneiderman, B. (1984b). *Review of Card, S. K., Moran, T. P., Newell, A.: The Psychology of Human-Computer Interaction*. Datamation, **30**, 236-240.
Shneiderman, B. (1986). Seven plus or minus two central issues in human-computer interaction. *Proceedings of the CHI '86 Conference on Human Factors in Computing Systems*. Boston.
Small, D. W., and Weldon, L. J. (1983). An experimental comparison of natural and structured query languages. *Human factors*, **25**, 253-263.
Smith, M. J. (1984). Health issues in VDT work. In J. Bennett, D. Case, Sandelin, J. and Smith, M. (eds.), *Visual Display Terminals: Usability Issues and Health Concerns* (p. 193-228). Englewood Cliffs: Prentice Hall.
Smith, M. J., Cohen, B. G. F., Stammerjohn, L. W., Jr. and Happ, A. (1981). An investigation of health complaints and job stress in video display operations. *Human Factors*, **23**, 387-400.
Smith, S. L. (1986). Standards versus guidelines for designing user interface software. *Behaviour and Information Technology*, **5**, 47-61.

Smith, S. L., and Mosier, J. N. (1984). The user interface to computer-based information systems: A survey of current software design practice. *Behaviour and Information Technology*, **3**, 195-203.
Spinas, P. (1984). Bildschirmeinsatz und psycho-soziale Folgen für die Beschäftigten. In Berufsverband Deutscher Psychologen (ed.), *Arbeit in moderner Technik* (pp. 503-516). Duisburg: Eigendruck.
Spinas, P. (1986). Analysis of dialogue systems: Field study. In E. Ulich (ed.), *Computer-aided Office Work* (pp. 9-16). Zürich: ETH, Lehrstuhl für Arbeits- und Organisationpsychologie.
Spinas, P. (in press). *VDU-work and user-friendly human-computer-interaction: Analysis of dialogue structures*. In M. Frese, E. Ulich and W. Dzida (eds.) *Psychological Issues of Human-Computer Interaction in the Work Place*. Amsterdam: North-Holland.
Spinas, P., Troy, N., and Ulich, E. (1983). *Leitfaden zur Einführung und Gestaltung von Arbeit mit Bildschirmsystemen*. München: CW-Publikationen.
Stammerjohn, L. W., Smith, M. J., and Cohen, B. G. F. (1981). Evaluation of work station design factors in VDT operations. *Human Factors*, **23**, 401-412.
Stark, L. W., and Johnston, P. G. (1984). Visual fatigue and VDT workplace. In J. Bennet, D. Case, J. Sandeli and M. Smith (eds.), *Visual Display Terminals: Usability Issues and Health Concerns* (pp. 229-270). Englewood Cliffs: Prentice Hall.
Stelovsky, J., and Sugaya, H. (1985). Command languages vs. menus or both? In H.-J. Bullinger (ed.), *Software-Ergonomie '85* (pp. 129-141). Stuttgart: Teubner.
Stevens, G. C. (1983). User-friendly computer systems? A critical examination of the concept. *Behaviour and Information Technology*, **2**, 3-16.
Streitz, N. (1985). Die Rolle von mentalen und konzeptuellen Modellen in der Mensch-Computer-Interaction: Konsequenzen für die Software-Ergonomie? In H.-J. Bullinger (ed.), *Software-Ergonomie '85* (pp. 280-292). Stuttgart: Teubner.
Sullivan, M. A., and Chapanis, A. (1983). Human factoring a text editor manual. *Behaviour and Information Technology*, **2**, 113-125.
Sundermeyer, P., and Haack-Vörsmann, L. (1983). On-board flight path planning as a new job concept for pilots. In G. Johannsen and J. E. Rijusdorp (eds.), *IFAC Analysis, Design, and Evaluations of Man-Machine Systems*. (pp. 197-203). Oxford: Pergamon.
Sydow, J. (1984). Sociotechnical change and perceived work situations. Some conceptual propositions and an empirical investigation in different office settings. *Office: Technology and People*, **2**, 121-132.
Sydow, J., Hattke, W., and Staehle, W. H. (1981). Situative Analyse der Bildschirmarbeit. *Zeitschrift für Organisation*, **50**, 215-223.
Szyperski, N. (1984). Das elektronische Büro. *Bild der Wissenschaft*, **1**, 73-98.
Taylor, F. W. (1911). *The Principles of Scientific Work*. New York: Harper and Row.
Thomas, J. C. and Schneider, M. L. (eds.) (1984). *Human Factors in Computing Systems*. Norwood: Ablex.
Troy, N. (1986). Designing attempt 1: Secretariats in a federal agency. In E. Ulich (ed.), *Computer-aided Office Work* (pp. 24-29). Zürich: ETH, Lehrstuhl für Arbeits- und Organisationspsychologie.
Tullis, T. S. (1985). Designing a menu-based interface to an operating system. *Proceedings of the CHI '85 Conference on Human Factors in Computing Systems* (pp. 79-84). San Francisco.
Turing, A. (1950). Computing machinery and intelligence. In E. A. Feigenbaum and J. Feldman (eds.), *Computers and Thought*. New York: McGraw Hill.
Turkle, S. (1984). *The Second Self: Computers and the Human Spirit*. NY: Simon and Schuster.
Turner, J. A., and Karasek, R. A. (1984). Software ergonomics: Effects of computer application design parameters on operator task performance and health. *Ergonomics*, **27**, 663-690.
Ulich, E. (1981). Möglichkeiten autonomieorientierter Arbeitsgestaltung. In M. Frese (ed.), *Stress im Büro* (pp. 159-177). Bern: Huber.

Ulich, E. (1983). Differentielle Arbeitsgestaltung—ein Diskussionsbeitrag. *Zeitschrift für Arbeitswissenschaft*, **37**, 12-15.
Ulich, E. (1985). Einige Anmerkungen zur Software-Psychologie. *Sysdata*, **10**, 53-58.
Ulich, E. (1986). Towards the design of user-oriented dialogue systems: Experiments. In E. Ulich (ed.), *Computer-aided Office Work* (pp. 6-9). Zürich: ETH, Lehrstuhl fuer Arbeits- und Organisationspsychologie.
Ulich, E., and Troy, N. (1986). Job organisation and allocation of functions between man and computer. In Klix, F. and Wandke, H. (eds.) *Man-Computer Interaction Research. MACINTER I* (pp. 421-427). Amsterdam: Elsevier, North Holland.
v. d. Veer, G., and Beishuizen, J. J. (undated). *Learning Styles in Conversation—A Practical Application of Pask's Learning Theory to Human-Computer Interaction*. Amsterdam: Manuscript.
v. d. Veer, G. C., Tauber, M. J., Green, T. R. G., and Gorny, P. (eds.) (1984). *Readings on Cognitive Ergonomics: Mind and Computers*. Berlin: Springer.
v. d. Veer, G., Tauber, M. J., Waern, Y., and v. Muylwijk, B. v. (1985). On the interaction between system and user characteristics. *Behaviour and Information Technology*, **4**, 289-308.
v. Muylwijk, B., v. d. Veer, G., and Waern, Y. (1983). On the implications of user variability in open systems. *Behaviour and Information Technology*, **2**, 313-326.
Volpert, W. (1981). *Sensumotorisches Lernen: Zur Theorie des Trainings in Industrie und Sport*. Frankfurt am Main: Fachbuchhandlung für Psychologie (3rd. ed.).
Volpert, W. (1985). *Zauberlehrlinge: Die gefährliche Liebe zum Computer*. Weinheim: Beltz.
Volpert, W. (1986). Gestaltbildung im Handeln: Zur psychologischen Kritik des mechanischen Weltbildes. *Gestalt Theory*, **8**, 43-60.
Volpert, W., Frommann, R., and Munzert, J. (1984). Die Wirkung allgemeiner heuristischer Regeln im Lernprozess—eine experimentelle Studie. Zeitschrift für Arbeitswissenschaft, **38**, 235-239.
Waern, Y. (1985). Learning computerized tasks as related to prior task knowledge. *International Journal of Man-Machine Studies*, **22**, 441-455.
Waern, Y., and Rabenius, L. (1985). *On the Role of Models in Instructing Novice Users of a Word Processing System*. Reprint: Department of Psychology Univ. of Stockholm.
Walton, R. E., and Vittori, W. (1983). New information technology: Organizational problem or opportunity? *Office: Technology and People*, **1**, 249-273.
Weinberg, G. M. (1971). *The Psychology of Computer Programming*. New York: Van Nostrand Reinhold.
Weizenbaum, J. (1976). *Computer Power and Human Reason: From Judgement to Calculation*. San Francisco: W. H. Freeman and Company.
Weltz, F. (undated). *Mitarbeiter-Befürchtungen und Managemetn-Fehler: Soziologische Aspekte bei der Einführung neuer Bürotechnologien*. Düsseldorf: Akzente.
Weltz, F. (1982). Arbeitsplatzgestaltung an Bildschirmarbeitsplätzen aus soziologischer Sicht. *AFA-Informationen*, **35**, 15-20.
Weltz, F., and Bollinger, H. (in press). *Mitarbeiterbeteiligung bei technisch-organisatorischen Verändergungen im Verwaltungsbereich*. Sozialwissenschaftliche Projektgruppe München.
Weltz, F. and Lullies, V. (1983). *Innovation im Büro: Das Beispiel Textverarbeitung*. Frankfurt: Campus.
Whiteside, J., Jones, S., Levy, P. S., and Wixon, D. (1985). User performance with command, menu, and iconic interfaces. *Proceedings of the CHI '85 Conference on Human Factors in Computing Systems* (pp. 185-191). San Francisco.
Wicklund, R. A. (1974). *Freedom and Reactance*. Hillsdale: Lawrence Erlbaum.
Widdel, H., and Kaster, J. (1985). Untersuchung zur formalen Transparenz eines Menüsystems. In H.-J. Bullinger (ed.), *Software-Ergonomie '85* (pp. 228-238). Stuttgart: Teubner.
Wiener, E. L. (1985). Beyond the sterile cockpit. *Human Factors*, **27**, 75-90.

Wilkins, A. J. (1985). Visual discomfort and cathode ray tube displays. In B. Shackel (ed.), *Human-Computer Interaction* (pp. 75-80). Amsterdam: Elsevier.

Williams, B. O., and Burch, J. L. (1985). *Human Foundations of Advanced Computing Technology: The Guide to the Select Literature*. Lawrence, Kansas: The Report Store.

Williges, R. C., Elkerton, J., Pittman, J. A., and Cohill, A. M. (1985). Providing online assistance to inexperienced computer users. In B. Shackel (ed.), *Human-Computer Interaction* (pp. 765-770). Amsterdam: Elsevier.

Williges, R. C., and Williges, B. H. (1983). Human-computer dialogue design considerations. In G. Johannsen and J. E. Rijusdorp (eds.), *IFAC Analysis, design, and evlauations of man-machine systems* (pp. 239-246.) Oxford: Pergamon.

Wixon, D., Whiteside, J., Good, M., and Jones, S. Building a user-defined interface. *Proceedings of the CHI '83 Conference on Human Factors in Computing Systems* (pp. 24-27). Boston.

Wohl, J. G. (1982). Information automation and the Apollo program: A retrospective. *IEEE Transactions on Systems, Man, and Cybernetics, SMC-12*, 469-478.

Wortman, C. B., and Brehm, J. W. (1975). Responses to uncontrollable outcomes: An integration of reactance theory and the learned helplessness model. In L. Berkowitz (ed.), *Advances in Experimental Social Psychology*. New York: Academic Press.

Wynne, B., and Otway, H. J. (1983). Information technology, power and managers. *Office: Technology and People*, **2**, 43-56.

Young, R. M. (1983). Surrogates and mappings: Two kinds of conceptual models for interactive devices. In D. Gentner and A. L. Stevens (eds.), *Mental Models* (pp. 35-52). Hillsdale: Erlbaum.

Zülch, G., and Starringer, M. (1984). Differentielle Arbeitsgestaltung in Fertigungen für elektronische Flachbaugruppen. *Zeitschrift für Arbeitswissenschaft*, **38**, 211-217.