Journal of Computer Assisted Learning (2000) **16**, 336-346

The effects of training wheels and self-learning materials in software training

M. Bannert
Department of General and Educational Psychology, University of Koblenz-Landau, Germany

Abstract The role of software training is becoming increasingly important due to the growing variety and complexity of modern software products. This paper focuses on the effectiveness and feasibility of two different kinds of individual learners' support in software training in classroom settings. This research question was investigated by analysing 11 text-processing courses which were conducted on the basis of a quasi-experimental research design. Firstly, the user interface was varied, i.e. half of the courses used the regular standard user interface of the software, whereas the other half used a training wheels interface in which all irrelevant functions were blocked. Secondly, in half of the courses all instruction was given by a human tutor whereas the participants in the other half received written self-learning material. A total of 72 university students majoring in different fields participated. Dependent measures were learning time, learning outcome, and learner satisfaction.

Keywords: Applications software; Experimental; Individual; Instruction; Self-learning materials; Training; Training wheels

Introduction

Today's applications software has become so complex that training must be considered an indispensable prerequisite for its effective usage. Presently, psychology-based guidelines for designing user training for modern software products are urgently needed. However, there are only a few psychological research studies addressing this topic. Hence, a research project was established to examine the topics of analysis, design, and evaluation of software training (Bannert, 1996). In the course of this work, software training was defined as a systematically planned teaching and learning process, the aim of which is to enable users to handle particular functions of an application software independently. Based on a theoretical framework linking research on human–computer interaction with instructional psychology, design recommendations for software training were derived and evaluated. The study reported here is one element of this research project which focuses on two different kinds of individual learners' support in software training in classroom settings: the user interface and the instructional media.

This paper is one of a selection of papers derived from a symposium at the EARLI conference in 1999 and from a poster session at the Deutsche Gesellschaft für Psychologie conference in 1998.

Correspondence: Maria Bannert, Department of General and Educational Psychology, University of Koblenz-Landau, Thomas-Nast-Strasse 44, 76829 Landau, Germany E-mail: bannert@rhrk.uni-kl.de

Previous research and research questions

The research of human–computer interaction (HCI) considers end-user training as a major topic of user support along with the design of help-systems and documentation (Shackel, 1991). Generally speaking, HCI-studies suggest three strategies for optimal computer-user training: First, training should include some exploration-supporting methods. Second, software training should be centred around the actual tasks of users and their working context. Finally, computer-supported tools facilitating the execution of such tasks as well as usable learning devices and help systems should be made available. Unfortunately, the majority of HCI-studies do not offer any detailed design recommendations on how to develop effective and efficient end-user training courses.

One exception is the research undertaken by Carroll and colleagues (Carroll, 1990, 1998) which was conducted in the context of the so-called *Minimalist Instruction Approach*. It calls for using real life settings with real user tasks, allowing users to get started quickly and to explore the system. Instruction is minimised as far as possible, and learners are supported in error recognition and recovery. Through these investigations Carroll has contributed to a theory of computer-user training not only by emphasising its importance for HCI-research, but also by delivering empirically validated design recommendations appropriate for user-training in real life settings.

One major type of technical support developed by Carroll are the so-called *training wheels*, i.e. a special user interface in which all irrelevant functions are blocked during the training session, thus reducing the demands on the learners' limited cognitive capacity. The objective is to make available exactly as much of the software functionality as is necessary so that learners can 'safely' practise its use. In several empirical studies, training wheels reduced learning time and, in addition, resulted in better learning outcomes and higher acceptance compared to the standard user interface (Carroll & Carrithers, 1984; Catrambone & Carroll, 1987). In these studies training wheels were developed for a menu-driven text processing system. Thus, the question of the current study is whether training wheels would also be superior in classroom training for modern software with windows–driven user interfaces using direct manipulation techniques.

One key issue discussed in Carroll's approach to user training is the role of exploratory learning, which he strongly supports as an appropriate learning strategy in software training. According to current educational psychology, self-controlled learning encompasses various aspects (e.g. Simons & De Jong, 1992; Simons,1992; Boekaerts, 1996), the most important being learners' regulation of learning activities according to their specific needs. In software training, learner's self-regulation can be realised by offering appropriate individual support, e.g. via paper-based or computer-based material. However, research reveals that learners often have problems controlling their learning activities, especially in the case of beginners.

Compared to the regulation of learning strategies or metacognitive skills, the management of learning time is much easier. In classroom training this could be easily realised by means of adequate training media, e.g. self-learning material instead of instruction by a tutor which is the common procedure today. When learning with step-by-step instructional material, learners are required to follow the described learning activities in a fixed sequence and so their control of learning activities is low. However, one does have control over learning time, which is much more difficult in tutor-led training. Hence, the second question of the study is whether the type of instructional delivery in classroom software training — i.e. human tutor-led instruction vs. written self-learning material — has an impact on course effectiveness and learners' satisfaction.

Method

Design

In a field study, all text-processing courses (WinWord 6.0™) held at the computer department of the University of Koblenz-Landau were investigated over a period of about one year. In the course of this, the research questions were addressed within a 2×2 quasi-experimental design without repeated measurement:

- Is the *Training Wheels Interface* superior to the application's standard interface?
- To what extent is the type of *instructional media* of importance, i.e. written self-learning material vs. human tutor?

With reference to the first question, two different user interfaces were offered in the classroom training courses: either the regular standard user interface (*STAND*) or a training wheels user interface (*TWINS*) in which all irrelevant functions were blocked. Whenever a function of the software that was not an objective in the training course was activated by a mouse-click, the shape of the mouse cursor changed into a small rectangle, signalling to the learner that this function was not available during the training session (Fach & Bannert, 1999). As far as the second question is concerned, instructions were given either by a human tutor (*TUTOR*) or by written self-learning material (*SLMAT*), although a human tutor was present. All other instructional aspects, such as learning objectives, sequencing of learning topics, context, and kind of practical activities, etc., were kept constant for all treatments.

Participants

Students. Data from 72 students (62 female and 10 male) majoring in different fields participating in 11 training courses were analysed (mean age of students was 25.74, *s.d.* = 5.09). Prior experience in text processing was required and this was checked in a short interview. In addition, participants were informed that the training courses were part of a research study involving measurements at the beginning and at the end of training. Each participant paid a DM 20.00 course fee.

Tutors. Each course was led by one of five research assistants, who were paid for the giving the course. They were all given extensive special training as preparation for the course. Retrospective analyses show that no instructor effects appeared with regard to course acceptance and learning task performance.

Course concept and materials

The development of the course concept was structured as an iterative process which allowed for an incremental refinement on the basis of expert ratings (see Bannert, 1998). The general aim of the course was to convey basic knowledge of WinWord 6.0 and to teach all necessary skills for students to be able to format seminar papers; this was evaluated by means of a criterion-referenced test, i.e. the learning performance task, at the end of the course. Consequently, the specification of the learning objectives, the learning contents, and their sequential order were based on the different activities entailed in writing a seminar paper with WinWord 6.0, for example creating and saving documents, editing and formatting text paragraphs, final checking and creating the overall layout. The course predominantly consisted of learning phases with direct instructions, which were designed on the basis of Klauer‘s teaching functions (Klauer, 1985). Nevertheless there was also a shift from direct guidance to guided exploration not only within the single lessons, but also over the entire training course. Explorative strategies were used especially during the exercises, where the participants had the chance to apply their knowledge at their individual pace. Error management took place during these exercises as well, but only with respect to functions belonging to a former or to the present lesson.

The entire course was composed of nine lessons, distributed over three days. Each session consisted of three lessons lasting from half an hour to one hour. In accordance with Klauer’s teaching functions each lesson started with an orientation about its specific contents. Following that, the particular functions were explained and their usage was demonstrated by formatting specific parts of a seminar paper. Each lesson ended with a summary, which was followed by an exercise in which students were required to apply the functions they had just learned in an exploratory manner. After each lesson there was a 15 minute break. In the human tutor groups the teaching of the basic WinWord-functions was done by the tutor in regular classroom settings which is typical of current software training. Only summaries and exercises were handed out on paper; on the basis of these papers students were able to individually explore and apply the lessons learned. In the self-learning material groups all instructions were delivered through the written material (Bannert, 1995), i.e. although participants were sitting together in the PC-room of the computer department, each individual learned at their own pace. The procedures followed in both the tutor-led and the self-learning material group were almost equal, for example all screen images presented in the written material were shown via overhead-foils in the tutor-led group, all examples, summaries and exercises were the same, and in all conditions students had their own PC.

Procedure

At the beginning of the WinWord-course, relevant *learner characteristics* (computer attitude, computer knowledge, experience in text-processing, interests in training course) were obtained by a questionnaire which took about 30 minutes to fill in. Following this, the training course began, which lasted 4 days with 4 hours of instruction per day. During the first three days the main learning sessions were held. At the beginning of the fourth day *learning outcome* as well as learner *satisfaction* were measured. Data were obtained by using logfile-recording (time data), a questionnaire (about 30 minutes to complete), and a criterion-referenced test, i.e. the learning performance task, which took about 70 minutes to carry out.

Toward the end of the course, all questions were answered which had been raised during the lessons and had not been dealt with at the time. Following this, students learned how to use the help system and WinWord-references, which took about two hours altogether. Finally, the students received a certificate for participating in the introductory WinWord-course.

Results

In the following, the data of all 72 participants distributed over 11 courses are analysed by 2*2 ANOVA analysis. Retrospective analyses show no treatment differences concerning the obtained learner characteristics, i.e. they were well balanced over the four treatment groups.

Learning time

On average, the compact courses lasted about 9½ hours ($m = 9.63$, $s.d. = 2.05$). Table 1 shows the average time needed for learning in each experimental condition which differs significantly. The greatest effect is due to the factor instructional media ($F_{1,68} = 25.62$, $MSE = 2.91$, $p < 0.001$), i.e. learning with self-learning material was about two hours faster ($m = 8.67$, $s.d. = 2.15$) than learning in the tutor-led, more traditional classroom format ($m = 10.63$, $s.d. = 1.35$).

This comparison must be considered carefully because students of the tutor-led groups had less control over their learning time since they had to adapt to the course pace, whereas students with the self-learning material could set their own pace and decide about the length of breaks between lessons. However, this does not lessen the fact that — on the whole — students of the self-learning material groups needed less time to learn the basics of WinWord. Different learning times raises the question whether the learning outcomes of the treatment groups would also differ.

Table 1. Time (in hours) needed to complete the training.

		STAND		TWINS	
		TUTOR (n = 19)	**SLMAT (n = 21)**	**TUTOR (n = 16)**	**SLMAT (n = 16)**
learning time	*m*	10.95	9.41	10.26	7.69
	s.d.	0.93	2.24	1.69	1.62

STAND = standard user interface; TWINS = Training Wheels with blocked functions;
TUTOR = tutor-led groups; SLMAT = self-learning material.

In addition, the factor user interface also showed a significant effect in respect to learning time ($F_{1,68} = 8.93$, $MSE = 2.91$, $p < 0.01$). Participants of the training wheels condition needed significantly less time ($m = 8.98$, $s.d. = 2.08$) than those of the courses with regular standard user interface ($m = 10.15$, $s.d. = 1.89$). Thus, blocking course-irrelevant functions reduced learning time, which is especially true in combination with self-instructional material; however, there was no significant interaction effect ($F_{1,68} = 1.65$, $MSE = 2.91$, $p > 0.05$). Again, the question arises as to whether different learning outcomes would be obtained.

Learning outcomes

At the end of the course, students were given a criterion-referenced test, i.e. the learning performance task, covering the material presented in the lessons. More specifically, they had to open an existing document and to edit, correct and format particular text paragraphs and tables. Finally, they had to check document layout, assign page numbers, add footnotes and set up a table of contents. Task completion took 71 minutes on average ($m = 70.88$, $s.d. = 17.68$) and did not differ significantly between the treatment groups. The task was performed on a WinWord document. Students saved the file which was subsequently analysed by means of an evaluation scheme to see if the required tasks were actually executed and, if so, of what quality. All files were judged by two independent raters, achieving 98.09% interrater agreement. On average the task was performed correctly in 89% of the cases ($m = 88.87\%$, $s.d. = 7.36$), which confirms the high effectiveness of the course concept. Table 2 presents the percentage of correctly solved tasks for each treatment.

Table 2. Percentage of correctly solved tasks.

		STAND		TWINS	
		TUTOR (*n* = 19)	**SLMAT** (*n* = 21)	**TUTOR** (*n* = 16)	**SLMAT** (*n* = 16)
learning outcome	*m*	85.16	92.60	87.50	89.75
	s.d.	8.16	4.57	8.23	6.43

STAND = standard user interface; TWINS = Training Wheels with blocked functions;
TUTOR = tutor-led groups; SLMAT = self-learning material.

A 2 x 2 ANOVA analysis revealed only a significant effect due to the factor 'instructional media' ($F_{1,68} = 8.71$, $MSE = 47.81$, $p < 0.01$). Students in the self-learning material groups showed significantly better performance ($m = 91.36$, $s.d. = 5.55$) than students of the tutor-led groups ($m = 86.23$, $s.d. = 8.16$). Thus, the newly developed material was not only efficient but also highly effective.

However, the variation of the user interfaces had no effect ($F_{1,68} = 0.02$, $MSE = 47.81$, $p > 0.05$), i.e. the training wheels interface was not superior ($m = 88.62$, $s.d. = 7.36$) compared to the standard interface ($m = 89.06$, $s.d. = 7.45$) as was expected according to Carroll's research. On the other hand, one must point out that since learning outcome was measured by using the regular standard user interface in all treatments, students learning with training wheels obviously had no problem switching over to the standard interface. The interaction effect was not significant ($F_{1,68} = 2.51$, $MSE = 47.81$, $p > 0.05$).

Learner satisfaction

At the end of the course, subjective course assessments were conducted with a questionnaire consisting of various statements about the course to be judged on a 6-point Likert scale. In the upper part of Table 3, the average assessments of course acceptance are listed (min. = 1 means no acceptance at all, max. = 6 means highest acceptance).

Table 3. Learners' satisfaction with the course and the training wheels.

		STAND		TWINS	
		TUTOR (*n* = 19)	**SLMAT** (*n* = 21)	**TUTOR** (*n* = 16)	**SLMAT** (*n* = 16)
		'All in all, I consider the course to be very good'			
Course acceptance	*m*	5.74	5.71	5.50	5.31
min = 1 'I do not agree at all'	*s.d.*	0.45	0.46	0.82	0.70
max = 6 'I completely agree'					
		'would be helpful'		*'was helpful'*	
Training wheels	*m*	1.57	1.33	3.81	2.44
min = 1 'I do not agree at all'	*s.d.*	0.64	0.58	1.60	1.21
max = 6 'I completely agree'					

STAND = standard user interface; TWINS = Training Wheels with blocked functions;
TUTOR = tutor-led groups; SLMAT = self-learning material;
High values mean high acceptance.

On the whole, the courses were judged very positively. A 2 x 2 ANOVA reveals a significant main effect of the factor user interface ($F_{1,68} = 4.85$, $MSE = 0.37$, $p < 0.05$). Thus, training wheels courses were judged not as good ($m = 5.41$, $s.d. = 0.76$) as courses using the standard user interface ($m = 5.73$, $s.d. = 0.45$), although the ratings were still quite positive. Since the factor instructional media had no significant effect ($F_{1,68} = 0.52$, $MSE = 0.37$, $p > 0.05$) and no significant interaction effect was obtained ($F_{1,68} = 0.32$, $MSE = 0.37$, $p > 0.05$) one can conclude that the usage of self-learning materials was not only efficient and effective but was also highly accepted by the learners.

In addition, the judgements shown in the lower part of Table 3 present further evidence that the training wheels interface was not accepted very well by the learners. Participants of training wheels condition were asked to judge how helpful the blocking of irrelevant functions really *was* by means of a 6-point Likert scale (min = 1, max = 6), whereas students of the standard interface condition were asked to imagine how helpful the blocking of all irrelevant functions *would be* in similar training courses (min = 1, max = 6).

In general, the blocking of irrelevant function was not seen as a supportive measure. Interestingly, groups who had experience with training wheels evaluated it not as negatively ($m = 3.13$, $s.d. = 1.56$) as the groups without experience, i.e. of the standard interface conditions ($m = 1.43$, $s.d. = 0.61$). Furthermore, the ratings of all participants of the training wheels courses were influenced by the factor instructional media ($F_{1,33} = 7.52$, $MSE = 2.01$, $p < 0.05$). Surprisingly, those students learning with self-learning material accepted the blocking to a significantly lesser degree ($m = 2.44$, $s.d. = 1.21$) than students of the tutor-led groups ($m = 3.81$, $s.d. = 1.6$). Especially for the self-learning conditions, the training wheels interface was expected to be a helpful support. However, in every course treatment there was always help available from the tutor if difficulties arose (e.g. loosing orientation in software). This may explain the finding that TWINS was not rated as an adequate means of user support in the self-learning material condition, since participants did not really perceive serious problems with the software because the tutor always came to their rescue.

Yet, considering the more positive ratings of the tutor-led group, it could be argued that participants judged TWINS' usefulness not in respect to their own individual learning but in regard to the classroom setting. As every software trainer will confirm, many problems occur in classroom training because of participants' clicking mistakes (e.g. inadvertently minimising the application windows, fading out important tool bars, etc.). In these training courses the tutor often had to interrupt the course because of such erroneous clicking in order to fix the user interface first and then proceed with the course. This could not happen with the blocked training wheels interface. Thus, TWINS was at least a helpful tool to avoid such disruptions, which tutors confirmed after the courses.

Further, there were other aspects which were assessed by the participants, i.e. course concept, examples and exercises, as well as learning materials (see Bannert, 1998). Overall, these aspects were rated positively with no treatment differences.

Discussion

To sum up, the research questions raised above can be answered on the basis of the data obtained as follows: in accordance with Carrolls' findings, participants from the 'training wheels' condition learned significantly faster compared to the participants of the 'standard user interface' treatment. However, in contradiction to Carrolls' assumptions (Carroll, 1990; 1998), they did not show better learning outcomes, which may be due to ceiling effects (see below) and moreover, their acceptance of the training wheels interface was not so high. In reference to the factor 'instructional media', students in the 'self-learning material' condition learned significantly faster and achieved significantly better learning outcomes than students in the 'human tutor' group. Both groups were highly satisfied with the course, i.e. there were no differences in learner overall course acceptance.

From an educational point of view, the results suggest that for software training in similar contexts, training wheels may not always be appropriate. The training wheels interface can only be recommended in cases with a tight time schedule. Detailed analysis of data revealed that learners experienced the blockage of irrelevant functions as a too restrictive approach. However, one must consider the fact that the participants of the training courses were always able to ask the human tutor for help. Therefore, a further experiment is currently being analysed to investigate whether the same results will be found in real self-learning conditions where no other person is present to help. Another important aspect which is worth examining is to let the learner decide whether to use the training wheels or the standard interface for different learning tasks. The written self-learning materials were highly effective. Again, this finding must be replicated under true self-learning conditions without any human support. However, one can conclude that at least for software training under similar conditions the use of such written learning material will lead to better results.

Yet, many questions remain unanswered. To begin with, the realisation of training wheels should be considered in more detail. Comparable to Carroll's description, the screen of TWINS was the same as for the regular user interface, the only difference was that the mouse cursor changed when clicking on a course-irrelevant function. Perhaps another type of implementation would be more effective and, in particular, would lead to greater learner acceptance. For example, Weinberg (1998; see also Leutner, this issue) developed an adaptive training wheels interface for CAD-training in which at the beginning of the course only a few functions were available and irrelevant functions were blocked by eliminating them from the interface. However, to avoid a completely different user interface the blocked and eliminated functions were replaced by dummies, thus the complexity of the training wheel interface was comparable to that of the standard interface. As the course progressed, more complex functions were presented on the screen. An experimental comparison showed significantly reduced learning time as well as better learning outcomes with the adaptive training wheels interface compared to the standard user interface. Unfortunately, learner satisfaction was not measured. All in all, other kinds of training wheels — especially as adaptable computer-supported learning environments for self-regulated software training — should be investigated in future research (Fach & Bannert, 1999).

With regard to the written learning material one must point out that — without doubt — it can be improved if the findings of current research of software documentation are considered (e.g. Odesalchi, 1986; Wright, 1988; Gong & Elkerton, 1990; Van der Meij & Lazonder, 1993; Van der Meij, this issue). In addition, cognitive load was not controlled for in the material, which also seems to have an impact in software training (e.g. Chandler & Sweller, 1996; Hagmann *et al.* 1998; Martin-Michiellot & Mendelsohn, this issue). What is probably most interesting from an educational point of view is the integration of information about metacognitive strategies in software learning material (e.g. Biemans & Simons, 1991). However, it was not possible to consider this psychological research in developing the material because of other constraints and research questions.

Some methodological critique might be formulated as well. First of all, experimental control is weak, not to say quite impossible, in a field study like this, i.e. experimental errors may occur which were not anticipated *a priori*, for example different group homogenity, group dynamics and learning atmosphere, etc. Although the study's internal validity may be low, one can assume that external validity is high, which was more important in this study. Further, in respect to the learning performance task, a ceiling effect may have occurred, since the percentage of solved tasks was very high; this fact makes it difficult to interpret the learning outcomes in regard to the treatment variation. The dilemma is that on one side, maximum course effectiveness was operationalised by 100% correct task solutions as measured by the criterion-referenced test, and the aim of the training measures was to achieve optimal performance rates, which was in fact realised. On the other side, this could also be interpreted as a ceiling effect. Presumably pretests at the beginning of the training could control for absolute performance increase of each treatment, however, participants will surely become very frustrated if they have to take a complex criterion-referenced test at the beginning of a course, especially in software training. This is the reason why such a pretest was not conducted.

In addition, only learning outcome was analysed but not the process of solving the task itself. It would be possible to conduct such an analysis by means of specific logfile recordings on which content-analytical procedures are applied. Moreover, a multidimensional evaluation allowing a simultaneous analysis of declarative and procedural knowledge, course acceptability, and specific learners' characteristics would be useful (Kraiger *et al.* 1993; Marsden & Funk-Müldner, 1993). Such an approach would make visible the impact of individual differences, for example, whether different levels of prior knowledge, computer attitudes, or interests might be connected with course effectiveness.

A major question in software training research is whether to employ explorative or directive methods in computer user training. Although this topic was not in the focus of this paper, the findings suggest that current critique of step-by-step instructions is no longer tenable. Rather the effectiveness of instructional methods is highly dependent on various psychological factors (Weinert & Helmke, 1995; Weinert, 1996). For example, Charney & Reader (1986) stated that in Carroll's studies explorative learning was confounded with problem solving, whereas it was not the case in the treatments of step-by-step instructions. Thus, they attribute the learning success more to problem-based learning than to exploration. In this respect, the presented study shows that guided instruction can yield a high degree of didactic effectiveness by an appropriate sequencing of the learning contents, for example by a stepwise presentation of typical problems which learners must solve immediately. This is also confirmed by a more recent study of Wiedenbeck and colleagues (this issue) which shows that the effectiveness of (exploration-based) software training is most likely determined by what activities the learner really undertakes during training.

Finally, from a psychological point of view, it is unrealistic to believe that the efficient use of complex application software can be learned within a four-day seminar. It is known that software applications have become so complex that training must be a difficult and long process. Thus, psychological research as well as the application of current psychological theories and know-how in order to facilitate learning and to provide support to learners in this domain still remains a challenge.

Acknowledgements

I would like to thank Peter W. Fach for implementing the Training Wheels interface and furthermore I want to thank him and Eleonore Hertweck for fruitful comments on former drafts of this paper. The Ministry of Education and Science of Rhineland-Palatinate [Bildung, Wissenschaft und Weiterbildung] is gratefully acknowledged for supporting this project in the scope of two research projects about software-training (Kap. 1515, and Kap. 1512).

References

Bannert, M. (1995) *Referate Gestalten Mit Word für Windows 6.0 – Ein Einführungskurs*. [Writing seminar papers with Word for Windows – an introduction] Empirische Pädagogik, Landau.

Bannert, M. (1996) *Gestaltung und Evaluation von EDV-Schulungsmaßnahmen. Eine empirische Studie zur Effektivität und Akzeptanz.* [Design and evaluation of computer training courses. An empirical study on effectivity and acceptance]. Empirische Pädagogik, Landau.

Bannert, M. (1998) Instructional Design for Problem-Based Learning: A Case Study in Computer User Training. In *Instructional Design for Problem-Based Learning* [Bi-annual Meeting of the SIG-ID in Maastricht, 26.-27. June, 1998.] (ed. J. Merrienboer & G. Moerkerke) pp. 75-98. University of Maastricht, Maastricht.

Biemans, H.J.A. & Simons, P.R.J. (1991) *Effekte der Einbettung von meta-kognitiven Fragen und Regulationshinweisen in ein computerunterstütztes Lernprogramm zur Textverarbeitung auf die Lernleistungen*. [Effects of the embedding of meta-cognitive questions and regulation hints in concurrent C.A.I. on the learning performances in word processing]. *Empirische Pädagogik*, **2**, 125–142.

Boekaerts, M. (1996) Self-regulated learning at the junction of cognition and motivation. *European Psychologist*, **1**, 100–112.

Carroll, J.M. & Carrithers, C. (1984) Blocking learner errors in a training wheels system. *Human Factors*, **26**, 377–389.

Carroll, J.M. (1990) *The Nurnberg Funnel. Designing Minimalist Instruction for Practical Computer Skill*. MIT Press, Cambridge, MA.

Carroll, J.M. (ed.) (1998) *Minimalism Beyond the Nurnberg Funnel*. MIT Press, Cambridge, MA.

Catrambone, R. & Carroll, J.M. (1987) Learning a word processor system with guided exploration and training wheels. In *Proceedings of CHI + GI'87. Human Factors in Computing Systems and Graphics Interface* [Toronto, April 5–9], (eds. J.M. Carroll & P.P. Tanner) pp. 169–174. Association for Computing Machinery, New York.

Chandler, P. & Sweller, J. (1996) Cognitive load while learning to use a computer program. *Applied Cognitive Psychology*, **10**, 151–170.

Charney, D.H. & Reader, L.M. (1986) Designing interactive tutorials for computer users. *Human–Computer Interaction*, **2**, 297–317.

Fach, P.W. & Bannert, M. (1999) Training wheels: an 'old' method for designing modern and adaptable learning environments. In *Proceedings of the 8th International Conference on Human–Computer Interaction* [Munich, August 22–27, 1999] (eds. H.J. Bullinger & J. Ziegler) pp. 725–729. Erlbaum, Mahwah, NJ.

Gong, R. & Elkerton, J. (1990) Designing minimal documentation using a GOMS model: A usability evaluation of an engineering approach. In *Proceedings of the CHI'90 Conference* (eds. J. Carrasco Chew & J. Whiteside) pp. 99–106. Association for Computing Machinery, New York.

Hagmann, S., Mayer, R. & Nenniger, P. (1998) Using structural theory to make a word-processing manual more understandable. *Learning and Instruction*, **8**, 19–35.

Klauer, K.J. (1985) Framework for a theory of teaching. *Teaching & Teacher Education*, **1**, 5–17.

Kraiger, K., Ford, J.K. & Salas, E. (1993) Application of cognitive, skill-based, and affective theories of learning outcomes to new methods of training evaluation. *Journal of Applied Psychology*, **78**, 311–328.

Marsden, N.F. & Funk-Müldnerer, K. (1993) Selbstgesteuertes Lernen am Computer – eine Zukunftsperspektive? Vergleich von konventioneller DV-Schulung und selbstgesteuertem Lernen am Computer. [Self-directed learning at the computer – a future perspective? Comparision of a conventional software training with self-controlled learning]. *Medienpsychologie*, **5**, 257–279.

Odesalchi, E. (1986) Documentation is the key to user success. *IEEE Transactions on Professional Communications*, **19**, 16–18.

Shackel, B. (1991) Whence and where – A short history of human–computer interaction. In *Human Aspects in Computing. Design and Use of Interactive Systems and Work with Terminals (18A)* (ed. H.J. Bullinger) pp. 3–18. Elsevier Science, Amsterdam.

Simons, P.R.J. & De Jong, F.P. (1992) Self-regulation and computer-assisted instruction. *Applied Psychology: an International Review*, **41**, 333–346.

Simons, P.R.J. (1992) Lernen,selbständig zu lernen – ein Rahmenmodell [Learning to learn – a framework]. In *Lern- und Denkstrategien* (eds. H. Mandl & H. Friedrich) pp. 251-264. Hogrefe, Göttingen.

Van der Meij, H. & Lazonder, A.W. (1993) Assessment of the minimalist approach to computer user documentation. *Interacting with Computers*, **5**, 355–370.

Weinberg, I. (1998) *Probleme von Softwaretrainings und Lösungsansätze: Instruktionspsychologisch begründete Entwicklung und Evaluation eines Double-Fading-Support-Konzepts für komplexe Softwaresysteme.* [Solving Problems of Software Training: Development of a Double-Fading Support Approach for Complex Software Systems which is Based on Instructional Psychology]. Shaker-Verlag, Aachen.

Weinert, F.E. & Helmke, A. (1995) Learning from wise mother nature or big brother instructor: The wrong choice as seen from an educational perspective. *Educational Psychologist*, **30**, 135–142.

Weinert, F.E. (1996) Lerntheorien und Instruktionsmodelle [Learning theories and instructional models]. In *Enzyklopädie der Psychologie,D/I (Pädagogische Psychologie: Psychologie des Lernens und der Instruktion, Band 2.* (ed. F.E. Weinert) pp. 1-48. Hogrefe, Göttingen.

Wright, P. (1988) Issues of content and presentation in document design. In *Handbook of Human–Computer Interaction* (ed. M. Helander) pp. 692–652. Elsevier Science, Amsterdam.