

# **Mensch Computer Kommunikation - Grundwissen 1**

Herausgegeben von Helmut Balzert



# **Einführung in die Software-Ergonomie**

Herausgegeben von  
Helmut Balzert, Heinz U. Hoppe,  
Reinhard Oppermann, Helmut Peschke,  
Gabriele Rohr, Norbert A. Streitz

Mit Beiträgen von  
David Ackermann, Helmut Balzert, Joachim Bauer,  
Edmund Eberleh, Thomas Hermann, Heinz U. Hoppe,  
Rolf Ilg, Reinhard Oppermann, Helmut Peschke,  
Gabriele Rohr, Thomas Schwab, Norbert A. Streitz,  
Eberhard Ulich, Jürgen Ziegler, Magdalena Zoepritz,  
Harald Zwerina



Walter de Gruyter · Berlin · New York 1988

*Herausgeber der Reihe*

Prof. Dr.-Ing. habil. Helmut Balzert  
Lehrstuhl für Software-Technik, Ruhr-Universität Bochum

*Bildnachweis*

Abb. 4-1 : I. T. T. A. GmbH, Köln  
Abb. 4-3 : Nixdorf Computer AG, Paderborn  
Abb. 4-6 : Hohe Electronics, Neunkirchen  
Abb. 4-7 : Hewlett-Packard, Bad Homburg  
Abb. 4-10: Macrotron, München  
Abb. 4-11: CalComp GmbH, Düsseldorf  
Abb. 4-13: TA Triumph-Adler AG, Nürnberg  
Abb. 4-14/15: Olivetti, Ivrea  
Abb. 4-15: frogdesign, Altensteig

Das Buch enthält 114 Abbildungen und 29 Tabellen.

*CIP-Kurztitelaufnahme der Deutschen Bibliothek*

**Einführung in die Software-Ergonomie** / hrsg. von Helmut Balzert . . . Mit Beitr. von David Ackermann . . . - Berlin ; New York : de Gruyter, 1988  
(Mensch-Computer-Kommunikation : Grundwissen ; 1)  
Mit 114 Abb. u. 29 Tab.  
ISBN 3-11-011939-0  
NE: Balzert, Helmut [Hrsg.]; Ackermann, David [Mitverf.]; Mensch-Computer-Kommunikation / Grundwissen

© Copyright 1988 by Walter de Gruyter & Co., Berlin 30.

Alle Rechte, insbesondere das Recht der Vervielfältigung und Verbreitung sowie der Übersetzung, vorbehalten. Kein Teil des Werkes darf in irgendeiner Form (durch Photokopie, Mikrofilm oder ein anderes Verfahren) ohne schriftliche Genehmigung des Verlages reproduziert oder unter Verwendung elektronischer Systeme verarbeitet oder verbreitet werden. - Printed in Germany.

Satz: H. Balzert, Witten. - Druck: Gerike GmbH, Berlin. - Bindearbeiten: Lüderitz & Bauer GmbH, Berlin. - Umschlagentwurf: Hansbernd Lindemann.

# Editorial

Computersysteme sind bereits heute aus dem Berufsleben nicht mehr wegzudenken. Im Jahr 2000 sollen sogar 90% aller Berufstätigen damit konfrontiert werden. Der menschengerechte Zugang zur Computertechnologie muß daher ein vordringliches Ziel von Forschung und Lehre sein. Neben der Gestaltung der Hardware spielt vor allem die **Software-Ergonomie** eine entscheidende Rolle.

Die Forschung hat in den letzten Jahren umfangreiche Ergebnisse zu diesem Gebiet hervorgebracht. Wichtige Erkenntnisse sind in der Buchreihe **Mensch-Computer-Kommunikation** publiziert.

Wegen der Relevanz dieser Thematik ist es nun an der Zeit, das wissenschaftlich abgesicherte Wissen in Form von Lehrbüchern einer breiteren Öffentlichkeit zur Verfügung zu stellen. Die Grundlagen der Software-Ergonomie sollten in Zukunft zum Basiswissen eines jeden gehören, der Softwaresysteme evaluiert und ihre Einführung in der Praxis vorbereitet, und erst recht zum Basiswissen eines jeden Systemanalytikers und Softwareentwicklers, der neue Softwaresysteme entwirft und realisiert. Damit ist das Ziel der neuen Buchreihe **Mensch-Computer-Kommunikation - Grundwissen** klar definiert.

Diese Buchreihe wendet sich an alle, die eine systematische Einführung in das Gebiet der Software-Ergonomie suchen. Das gilt gleichermaßen für **Studenten** und **Praktiker**. Die Probleme der Software-Ergonomie lassen sich nur durch interdisziplinäre Zusammenarbeit von Spezialisten verschiedener Fachrichtungen lösen. Diese Buchreihe gehört daher in die Hand eines jeden **Informatikers, Arbeitswissenschaftlers** und **Psychologen**.

Der vorliegende erste Band *Einführung in die Software-Ergonomie* der Buchreihe **Mensch-Computer-Kommunikation - Grundwissen** bietet eine systematische, interdisziplinäre Einführung in dieses Thema.

Das Buch entstand im Rahmen der von mir 1986 initiierten Software-Ergonomie-Herbstschule (SEH) der Gesellschaft für Informatik. Diese Herbstschule findet seitdem jährlich einmal statt. Die Autoren des vorliegenden Buches stammen aus verschiedenen Disziplinen und forschen auf dem Gebiet der Software-Ergonomie. Ihre Erfahrungen vermitteln sie auf der Herbstschule.

Die Herausgeber dieses Buches haben die Erfahrungen der vergangenen Herbstschulen in ein didaktisch-methodisches Konzept eingebracht, deren Ergebnis hier vorliegt.

Möge dieses Buch Ihnen als Leser einen fundierten Einblick in das noch junge Gebiet der Mensch-Computer-Kommunikation geben und Sie dazu anregen, bei der ergonomischen Gestaltung, Entwicklung und Evaluation von Software-Systemen mitzuwirken.

Helmut Balzert

Herausgeber der Reihe Mensch-Computer-Kommunikation - Grundwissen



# Vorwort

Computersysteme sind auf dem Wege, in allen Bereichen unserer Gesellschaft eine zentrale Rolle zu spielen. Sie werden überall vorhanden sein und dabei nicht nur das Arbeitsleben entscheidend prägen, sondern auch weite Bereiche des privaten Informations- und Kommunikationsverhaltens beeinflussen. Obwohl eine Reihe von Computersystemen so ausgelegt sind, daß sie ihre Funktion im wesentlichen autonom ausführen, erfordert die Mehrzahl einen Zugang für den Benutzer, d.h. eine Möglichkeit der direkten Interaktion zwischen Benutzer und Computersystem: die Mensch-Computer-Schnittstelle. Es ist mittlerweile wohl unumstritten, daß die Verwendungsfähigkeit von Mensch-Computer-Systemen maßgeblich durch die Gestaltung der Mensch-Computer-Schnittstelle (MCS) bestimmt wird. Das Ausmaß an ergonomischer und menschengerechter Gestaltung entscheidet oft über den Erfolg oder Mißerfolg von Systemen. Außerdem ist der Anteil der MCS am Gesamtprogrammieraufwand eines Systems ein nicht zu unterschätzender Kostenfaktor. So zeigen Studien für ausgewählte Anwendungen im kaufmännischen Bereich, daß die Programmierung der MCS im Mittel 59 % des gesamten Programms ausmachte. Neuere Untersuchungen besagen, daß die Realisierung der MCS in vielen Fällen zwischen 30 % und 50 % des Codes von wissensbasierten oder Expertensystemen beträgt.

Obwohl die MCS also wichtig zu sein scheint, ist sie in mancherlei Hinsicht immer noch eine der am wenigsten verstandenen und in vielen Fällen am schlechtesten gestalteten Komponenten von Mensch-Computer-Systemen. Die Gründe dafür sind vielfältig und werden in verschiedenen Teilen dieses Buches ausführlich behandelt. Einer der zentralen Gründe ist sicher auch der Mangel an ergonomischem Wissen bei denjenigen, die an der Entwicklung von Systemen und da speziell der MCS beteiligt sind. Verbindet sich Unkenntnis mit Unverständnis und fehlendem Problembewußtsein für die spezifische Komplexität und Problematik der Mensch-Computer-Interaktion, dann bekommen wir die Mensch-Computer-Schnittstellen, wie wir sie oft leidvoll erfahren müssen.

Das vorliegende Buch ist aus der Notwendigkeit entstanden, diese Lücken in der augenblicklichen Ausbildung von Informatikern und anderen an der Systementwicklung Beteiligten zu schließen. Es soll helfen, das existierende Defizit an problemspezifischem Wissen zu verringern und Anstöße zum Überdenken herkömmlicher Vorgehensweisen geben. Das Buch nennt sich *Einführung in die Software-Ergonomie* und bezieht sich damit auf ein noch relativ junges Forschungsgebiet, das sich der oben skizzierten Fragestellungen angenommen hat.

Dieses Buch ist aus einer jährlich stattfindenden einwöchigen Weiterbildungsveranstaltung entstanden, die die Fachgruppe *Software-Ergonomie* der Gesellschaft für Informatik in Verbindung mit der Deutschen Informatik-Akademie 1986 das erste Mal durchgeführt hat: die Software-Ergonomie Herbstschule (SEH). Im Rahmen der Vorbereitung und Durchführung der SEH '87 ist dann vom Programmkomitee eine didaktische Konzeption für ein entsprechendes Lehrbuch entwickelt

worden. Das vorliegende Buch basiert auf Beiträgen der Referenten zur SEH '87, die von den Autoren gemäß der von den Herausgebern entwickelten Konzeption für die Buchversion überarbeitet wurden. Es handelt sich dabei mit Absicht um keinen Tagungsband, in dem die neuesten, vielleicht noch spekulativen Überlegungen aus der Forschungsarbeit berichtet werden. Vielmehr haben wir einerseits versucht, eine Übersicht über die grundlegenden Konzepte und Methoden zu geben, die dieses interdisziplinäre Gebiet prägen. Andererseits werden diese ergänzt durch Ergebnisse fundierter experimenteller Untersuchungen – soweit diese in einem noch so jungen und sich dynamisch entwickelnden Gebiet wie der Software-Ergonomie schon vorhanden sein können. Dieses Buch ist neben der individuellen Aus- und Weiterbildung vor allem auch für den Einsatz in Lehrveranstaltungen verschiedener Fachrichtungen zu dieser Thematik gedacht.

Die Beiträge in diesem Buch sind in vier Hauptgruppen geordnet, die durch ein einführendes Kapitel und einen Ausblick eingerahmt werden. Im ersten Kapitel werden die Fragestellungen der Software-Ergonomie, die beteiligten Disziplinen und die dabei verwendeten Forschungsstrategien vorgestellt, verbunden mit einem historischen Überblick über die Hardware- und Softwareentwicklung.

Es folgen in der Gruppe *Grundlagen* drei Kapitel, die Erkenntnisse über die menschliche Informationsverarbeitung (Gedächtnis, Wahrnehmung, Wissensrepräsentation, kognitive Prozesse), die Einordnung der MCS-Gestaltung in Arbeits- und Organisationsstrukturen und soziotechnische Systemgestaltung, sowie die technischen Grundlagen und Voraussetzungen zur Gestaltung der MCS (Ein-/Ausgabegeräte wie z. B. Tastaturen und Zeigeeinstrumente) vermitteln.

In der aus vier Kapiteln bestehenden Gruppe *Dialogformen* werden nach der Analyse und Klassifikation von Dialogformen die wichtigsten Dialogformen im einzelnen detailliert dargestellt. Dabei handelt es sich um die Menüauswahl als eine insbesondere für Anfänger und gelegentliche Benutzer geeignete Interaktionsform, die eher für fortgeschrittene Benutzer geeigneten Interaktionssprachen wie Kommandos, Abfrage- und Programmiersprachen, gefolgt von Empfehlungen für die Gestaltung von Masken und Formularen. Den Abschluß dieser Gruppe bildet das Kapitel über direkte Manipulation, einer Interaktionsform, die von neuen technischen Möglichkeiten Gebrauch macht.

In der Gruppe *Hilfesysteme* gibt es zwei Kapitel, von denen das erste die konstruktiven Voraussetzungen behandelt, während das zweite sich kritisch mit dem Einsatz, insbesondere von aktiven und adaptiven Hilfesystemen auseinandersetzt.

Die Gruppe *Entwurf und Gestaltung* besteht aus fünf Kapiteln. Sie wird eingeleitet durch einen Beitrag zum Verhältnis von Aufgabenanalyse und Funktionsentwurf, bezieht sich damit über die MCS hinaus auf die Funktionalität. Es folgen Ausführungen zur Einbindung software-ergonomischer Aspekte in den Software-Entwicklungsprozeß, u. a. unter Berücksichtigung von Normen. Daran schließt sich ein Beitrag zum Einsatz von Werkzeugen bei der Prototypentwicklung von MCS an, verbunden mit einem Überblick über Programmiersprachen und Entwicklungsumgebungen. Die Beteiligung der zukünftigen Benutzer am Systementwicklungsprozeß und geeignete Maßnahmen zur Einführung von neuen Systemen sind Gegenstand des vierten Beitrags in dieser Gruppe, die abgeschlossen wird durch Ausführungen zu Evaluationsverfahren von Softwaresystemen, insbesondere vor dem

Hintergrund von Richtlinien und Normen, wie sie z.Zt. in der Diskussion sind. Im Kapitel *Trends und Perspektiven* wird der Stand der gegenwärtigen Situation in Hinblick auf neue technische Entwicklungen, aber auch auf den Einsatz neuer Konzepte für interaktive, wissensbasierte Systeme extrapoliert.

Das vorliegende Buch ist eine Einführung und erhebt nicht den Anspruch, für alle anstehenden Probleme der MCS-Gestaltung eine Lösung anbieten zu können, zumal es auch in der Forschung noch eine Reihe von offenen Fragen gibt. Da es aber z. Zt. keine adäquate Ausbildung in Software-Ergonomie gibt, war es notwendig, das sehr verstreut und teilweise nur fachspezifisch formulierte Wissen zusammenzutragen und in einer aufbereiteten Form zusammenzustellen. Wir hoffen, daß die Leser dieses Buches davon für ihre Arbeit profitieren werden, in welcher Weise sie auch immer – als Benutzer oder als Entwickler von interaktiven Systemen – mit dieser Thematik konfrontiert sind.

Die Herausgeber



# Inhalt

## **Einführung**

<b>1 Fragestellungen und Forschungsstrategien der Software-Ergonomie . . . . .</b>	<b>3</b>
<i>Norbert A. Streitz</i>	
1.1 Ausgangspunkt . . . . .	3
1.2 Sachproblem und Interaktionsproblem . . . . .	6
1.3 Forschungsziele und Systementwurf . . . . .	9
1.4 Hardware- und Software-Entwicklung im Rückblick . . . . .	10
1.5 Die Wissenschaftslandschaft . . . . .	13
1.6 Beteiligte Disziplinen und Forschungsstrategien . . . . .	14
1.7 Schlußbemerkung . . . . .	20

## **Grundlagen**

<b>2 Grundlagen menschlicher Informationsverarbeitung . . . . .</b>	<b>27</b>
<i>Gabriele Rohr</i>	
2.1 Einführung . . . . .	27
2.2 Visuelle Wahrnehmung . . . . .	28
2.2.1 Foveales Sehen und Blickbewegungen . . . . .	28
2.2.2 Gestaltgesetze . . . . .	31
2.3 Aufmerksamkeitssteuerung . . . . .	33
2.4 Gedächtnisorganisation . . . . .	36
2.4.1 Struktur des Gedächtnisses . . . . .	36
2.4.2 Codierung und Abrufprozesse . . . . .	36
2.5 Wissensrepräsentation . . . . .	39
2.5.1 Kategorienbildung . . . . .	39
2.5.2 Verknüpfung von Wissenselementen . . . . .	41
2.6 Kognitive Prozesse . . . . .	43
2.6.1 Handlungssteuerung . . . . .	43
2.6.2 Erwerb neuer Wissenssysteme: Metaphern . . . . .	44
2.7 Methoden der Datenerhebung . . . . .	46
2.7.1 Einfache Reaktionen . . . . .	46
2.7.2 Reaktionsmuster . . . . .	46
<b>3 Arbeits- und organisationspsychologische Aspekte . . . . .</b>	<b>49</b>
<i>Eberhard Ulich</i>	
3.1 Kriterien zur Bewertung von Arbeits- und Organisationsstrukturen . . . . .	49
3.2 Strategien der Arbeitsgestaltung . . . . .	49
3.3 Das Konzept der soziotechnischen Systemgestaltung . . . . .	50
3.4 Technologie als Option . . . . .	51
3.5 Tätigkeitsspielraum und Aufgabenorientierung . . . . .	52
3.5.1 Zum Konzept des Tätigkeitsspielraums . . . . .	52
3.5.2 Merkmale der Aufgabengestaltung . . . . .	55
3.6 Differentielle Arbeitsgestaltung . . . . .	56

- 3.7 Kriterien der Benutzerfreundlichkeit . . . . . 57
  - 3.7.1 Flexibilität / Individualisierbarkeit . . . . . 59
  - 3.7.2 Partizipation . . . . . 61
- 4 E/A-Geräte für die Mensch-Computer-Interaktion . . . . . 67
  - Helmut Balzert*
    - 4.1 Einleitung . . . . . 67
    - 4.2 Tastaturen . . . . . 68
      - 4.2.1 Funktionstasten . . . . . 71
      - 4.2.2 Cursortasten . . . . . 71
      - 4.2.3 Weiterentwicklungen . . . . . 72
    - 4.3 Zeigeinstrumente . . . . . 73
      - 4.3.1 Direkte Manipulation . . . . . 73
      - 4.3.2 Kriterien für Zeigeinstrumente . . . . . 74
      - 4.3.3 Der Lichtgriffel . . . . . 75
      - 4.3.4 Berührungsempfindliche Gebiete, Flächen und Folien . . . . . 76
      - 4.3.5 Die Maus . . . . . 78
      - 4.3.6 Der Steuerknüppel . . . . . 79
      - 4.3.7 Die Rollkugel . . . . . 81
      - 4.3.8 Das Grafiktablett . . . . . 82
      - 4.3.9 Vergleich von Zeigeinstrumenten . . . . . 83
    - 4.4 Handschrifteingabe . . . . . 85
    - 4.5 Sprachspeicherung und Spracherkennung . . . . . 86
    - 4.6 Bildverarbeitung . . . . . 89
    - 4.7 Bildschirme . . . . . 89
    - 4.8 Drucker . . . . . 91
    - 4.9 Kombinierte E/A-Geräte . . . . . 91
- Dialogformen**
  - 5 Klassifikation von Dialogformen . . . . . 101
    - Edmund Eberleh*
      - 5.1 Definition von Dialogform . . . . . 101
      - 5.2 Grundlagen der Klassifikation . . . . . 102
        - 5.2.1 Ziel und Probleme der Klassifikation . . . . . 102
        - 5.2.2 Bestimmung und Reduktion des Merkmalsraumes . . . . . 103
      - 5.3 Charakterisierende Merkmale von Dialogformen . . . . . 103
        - 5.3.1 Komponenten und Determinanten der Kommunikation . . . . . 104
        - 5.3.2 Prozesse und Zyklus der Interaktion . . . . . 104
        - 5.3.3 Medium der Kommunikation . . . . . 106
      - 5.4 Integration verschiedener Dialogformen: ein Beispiel . . . . . 111
      - 5.5 Ansätze zur Klassifikation von Dialogformen . . . . . 113
        - 5.5.1 Eindimensionale Klassifikationen . . . . . 113
        - 5.5.2 Mehrdimensionale Klassifikationen . . . . . 116

<b>6</b>	<b>Menüauswahl</b>	<b>121</b>
	<i>Edmund Eberleh</i>	
6.1	Ein alltägliches Beispiel	121
6.2	Allgemeine Eigenschaften von Menüauswahlen	122
6.2.1	Konzeptuelle Charakteristika	122
6.2.2	Vor- und Nachteile	122
6.3	Strukturelle Organisation	123
6.3.1	Einzelne Menüs	123
6.3.2	Darstellung der Items	126
6.3.3	Lineare Sequenzen	126
6.3.4	Baumstrukturen (Hierarchien)	127
6.3.5	Netzwerke (Heterarchien)	128
6.4	Auswahlkürzel	128
6.4.1	Ziffern	129
6.4.2	Buchstaben	129
6.4.3	Ziffern plus Buchstaben	130
6.4.4	Zeigen	130
6.5	Gestaltung von Menüs	131
6.5.1	Benennung der Titel	131
6.5.2	Benennung der Items	132
6.5.3	Anordnung der Items	132
6.5.4	Graphische Gestaltung	132
6.5.5	Orientierungshilfen für Benutzer	133
6.6	Antwort- und Bildaufbauzeit	134
6.7	Beschleunigung der Menüwahlen	134
6.7.1	Pfadnamen	135
6.7.2	Direkter Zugriff	136
6.7.3	Makro-Kommandos	136
<b>7</b>	<b>Interaktionssprachen</b>	<b>139</b>
	<i>Magdalena Zoepritz, Gabriele Rohr</i>	
7.1	Einführung	139
7.1.1	Definition und Abgrenzung	139
7.1.2	Eigenschaften von Interaktionssprachen	140
7.2	Die vier Hauptgruppen von Interaktionssprachen	141
7.2.1	Kommandosprachen	141
7.2.2	Abfrage-Sprachen	144
7.2.3	Programmiersprachen	147
7.2.4	Makros	148
7.3	Hilfsmittel für das Lernen und Behalten von Interaktionssprachen	149
7.3.1	Wo ist welche Hilfe notwendig	149
7.3.2	Untersuchungen zu Benennungen und mnemonischen Abkürzungen in Kommandosprachen	151
7.3.3	Untersuchungen zu natürlichsprachlichen Hilfsmitteln in Abfragesprachen	152
7.3.4	Untersuchungen zur Strukturierung von Befehlsfolgen in Programmiersprachen	155

7.4	Schlußbemerkung . . . . .	158
8	Masken und Formulare . . . . .	163
	<i>Harald Zwerina</i>	
8.1	Dialogsysteme in Maskentechnik . . . . .	163
8.2	Situationsanalyse im Vorfeld der Entwicklung . . . . .	164
8.3	Gestaltung von Dialoginhalt und Dialogablauf . . . . .	166
8.4	Visuelle Gestaltung der Maskeninformation . . . . .	170
9	Direkte Manipulation . . . . .	175
	<i>Rolf Ilg, Jürgen Ziegler</i>	
9.1	Einführung . . . . .	175
9.2	Definitionen der Direkten Manipulation . . . . .	175
9.2.1	Beschreibung der Direkten Manipulation nach Shneiderman . . . . .	175
9.2.2	Hutchins, Hollan & Norman: Direkt manipulative Benutzerschnittstellen . . . . .	177
9.2.3	Direkte Manipulation in einem Modell der Mensch-Rechner-Interaktion . . . . .	178
9.2.4	Direktheit . . . . .	181
9.3	Eigenschaften und Anwendungen von Direkter Manipulation . . . . .	183
9.3.1	Funktionsaufruf . . . . .	183
9.3.2	Syntax . . . . .	186
9.3.3	Generische Kommandos . . . . .	187
9.3.4	Integration funktionaler Bereiche . . . . .	188
9.3.5	Funktionen und Attributierungen . . . . .	189
9.4	Empirische Untersuchungen . . . . .	191
	<b>Hilfesysteme</b>	
10	Anforderungen an Hilfesysteme . . . . .	197
	<i>Joachim Bauer, Thomas Schwab</i>	
10.1	Einleitung . . . . .	197
10.2	Die Notwendigkeit für Hilfesysteme . . . . .	197
10.3	Klassifikation von Hilfesystemen . . . . .	198
10.4	Anforderungen an Hilfesysteme . . . . .	199
10.4.1	Ein guter Entwurf des Anwendungssystems ist Voraussetzung . . . . .	199
10.4.2	Hilfesysteme müssen einfach zu bedienen sein . . . . .	200
10.4.3	Dynamische Hilfe ist notwendig . . . . .	200
10.4.4	Individuelle Hilfe ist notwendig . . . . .	201
10.4.5	Aktive Hilfe ist notwendig . . . . .	202
10.4.6	Das Gesamtsystem muß wissensbasiert sein . . . . .	203
10.4.7	Bewertung der Forderungen . . . . .	204
10.5	Ein integriertes Hilfesystem . . . . .	205
10.5.1	Die zentrale Komponente COMMANDHELP . . . . .	205
10.5.2	Die aktive Komponente AKTIVIST . . . . .	207
10.5.3	Die natürlichsprachliche Komponente PASSIVIST . . . . .	208

10.5.4	Die dynamische Komponente DYNHELP . . . . .	210
10.5.5	Bewertung des Hilfesystems . . . . .	211
10.6	Ausblick . . . . .	212
11	Probleme bei der Konstruktion und beim Einsatz von Hilfesystemen . . .	215
	<i>Thomas Hermann</i>	
11.1	Einleitende Systematisierung . . . . .	215
11.2	Formen der Hilfestellung . . . . .	217
11.2.1	Zur Orientierung von Erklärungen . . . . .	217
11.2.2	Zur Form der Darstellung von Hilfen . . . . .	218
11.2.3	Formen informationstechnisch-gebundener Hilfestellung und Systemerklärung . . . . .	220
11.3	Probleme bei der Kontextberücksichtigung . . . . .	222
11.4	Probleme beim Einsatz von wissensbasierten Hilfesystemen . . . .	225
11.5	Abschließende Einsichten . . . . .	226
<b>Entwurf und Gestaltung</b>		
12	Aufgabenanalyse und Funktionsentwurf . . . . .	231
	<i>Jürgen Ziegler</i>	
12.1	Einleitung . . . . .	231
12.2	Grundbegriffe der Aufgabenanalyse . . . . .	232
12.2.1	Was ist eine Aufgabe? . . . . .	232
12.2.2	Aufgabenmerkmale . . . . .	233
12.2.3	Ebenen der Aufgabenanalyse . . . . .	234
12.2.4	Aufgabenangemessenheit . . . . .	234
12.3	Analyse- und Entwurfsmethoden . . . . .	235
12.3.1	Analyseorientierte Verfahren . . . . .	236
12.3.2	Entwurfsorientierte Verfahren . . . . .	237
12.3.3	Entwurf von Objekten und Operationen (Konzeptueller Entwurf) . . . . .	238
12.3.4	Optimierungsfragen beim semantischen Entwurf . . . . .	241
12.3.5	Kognitive Aufgabenanalysen in der Mensch-Rechner- Interaktion . . . . .	242
12.3.6	Task-Action-Grammatiken (TAG) . . . . .	243
12.3.7	GOMS-Modelle und die Theorie der kognitiven Komplexität . . . . .	246
12.4	Schlußbemerkung . . . . .	250
13	Empirie des Softwareentwurfs: Richtlinien und Methoden . . . . .	253
	<i>David Ackermann</i>	
13.1	Einleitung . . . . .	253
13.2	Aspekte und Schritte des Softwareentwicklungsprozesses . . . . .	253
13.3	Normen, Richtlinien und Leitfaden . . . . .	255
13.4	Der Leitfaden zur Einführung und Gestaltung von Arbeit mit Bildschirmssystemen . . . . .	256
13.4.1	Gliederung des Leitfadens und Beispiele . . . . .	257

13.4.2	Ausschnitte aus (Arbeits-)Psychologische Aspekte der Bild- und Dialoggestaltung . . . . .	257
13.4.3	Diskussion . . . . .	259
13.5	Deutsche Industrie Norm: DIN 66234, Teil 8 . . . . .	260
13.5.1	Überblick und Beispiel . . . . .	260
13.5.2	Diskussion . . . . .	261
13.6	Entwurfsrichtlinien von Smith und Mosier . . . . .	262
13.6.1	Beispiele . . . . .	262
13.6.2	Diskussion . . . . .	264
13.7	Methoden der Entwicklung . . . . .	264
13.7.1	Analyse des Handlungsspielraums . . . . .	264
13.7.2	Analyse der in den Dialog abzubildenden Aufgabenstellung . . . . .	265
13.7.3	Definition der Anforderungen . . . . .	265
13.7.4	Fallbeispiel 1: Aspekte des iterativen Entwurfs . . . . .	266
13.7.5	Methoden des Entwurfs . . . . .	267
13.7.6	Implementation . . . . .	267
13.7.7	Benutzerbeteiligung . . . . .	269
13.7.8	Diskussion . . . . .	269
13.8	Fallbeispiel 2: Dialogstruktur und Kompetenzentwicklung . . . . .	269
13.8.1	Aufgabenstellung . . . . .	270
13.8.2	Experimente zum Dialogentwurf . . . . .	270
13.8.3	Dialogstruktur und Wissen . . . . .	271
13.8.4	Diskussion . . . . .	272
13.9	Ausblick . . . . .	273
14	Werkzeuge für die Prototypenentwicklung von Benutzerschnittstellen . . . . .	277
	<i>Heinz Ulrich Hoppe</i>	
14.1	Prototypentwicklung von Benutzerschnittstellen . . . . .	277
14.2	Prototypenentwicklung als Modellkonstruktion – verschiedene Ansätze und Sichtweisen . . . . .	278
14.3	Schnittstellengeneratoren und UIMS . . . . .	280
14.3.1	Allgemeine Einordnung . . . . .	280
14.3.2	Das System SYNICS . . . . .	281
14.3.3	Das System XS-2 . . . . .	283
14.4	Werkzeuge und Methoden aus dem Bereich der KI . . . . .	285
14.4.1	Überblick . . . . .	285
14.4.2	Sprachen und Programmierstile der KI . . . . .	286
14.5	Programmierungsumgebungen auf LISP-Maschinen . . . . .	291
14.6	Vergleichende Gegenüberstellung verschiedener Werkzeuge . . . . .	294
15	Partizipative Entwicklung und Einführung von Informationssystemen . . . . .	299
	<i>Helmut Peschke</i>	
15.1	Einleitung . . . . .	299
15.2	Ganzheitliche Gestaltung . . . . .	301
15.3	Rückgekoppelte Vorgehensweise . . . . .	304
15.4	Beteiligung der Betroffenen . . . . .	307

15.4.1	Beteiligungsform . . . . .	308
15.4.2	Ausprägungen der Partizipation . . . . .	309
15.4.3	Inhalte der Partizipation . . . . .	311
15.5	Einordnung in der Praxis . . . . .	314
15.5.1	Beispiel einer Beteiligung im Rahmen von Organisationsentwicklung . . . . .	314
15.5.2	Beispiel korrekativer Software-Ergonomie . . . . .	317
15.6	Zusammenfassung und Ausblick . . . . .	319
16	Software-ergonomische Evaluationsverfahren . . . . .	323
	<i>Reinhard Oppermann</i>	
16.1	Einleitung . . . . .	323
16.2	Unbestimmtheit software-ergonomischer Evaluationskriterien . . . . .	323
16.3	Ebenen von Systemevalationen . . . . .	325
16.4	Evaluation in unterschiedlichen Phasen der Systementwicklung . . . . .	326
16.5	Methoden der Evaluation . . . . .	327
16.5.1	Subjektive Evaluationsmethoden . . . . .	327
16.5.2	Objektive Evaluationsmethoden . . . . .	328
16.5.3	Leitfadenorientierte Evaluationsmethoden . . . . .	329
16.5.4	Experimentelle Evaluationsmethoden . . . . .	330
16.6	Exemplarischer Evaluationsansatz EVADIS . . . . .	331
16.6.1	Ziel und Entwicklungsstand des Leitfadens . . . . .	332
16.6.2	Prüf-Items . . . . .	332
16.6.3	Standardaufgabe . . . . .	334
16.6.4	Durchführungsvorschrift . . . . .	335
16.6.5	Bezugsrahmen der Itemsammlung . . . . .	337
16.6.6	Auswertung nach technischen Systemkomponenten und/ oder software-ergonomischen Prüfkriterien . . . . .	339
16.7	Schlußbemerkung . . . . .	340
	<b>Ausblick</b>	
17	Trends und Perspektiven der Software-Ergonomie . . . . .	345
	<i>Helmut Balzert</i>	
17.1	Einführung . . . . .	345
17.2	Ergonomische Organisations-Gestaltung . . . . .	346
17.3	Ergonomische Gestaltung von Anwendungssystemen . . . . .	347
17.4	Auskunfts- und Beratungssysteme . . . . .	350
17.5	Adaptierbare und adaptive Systeme . . . . .	352
17.6	Konstruktion ergonomischer Software-Systeme . . . . .	358
17.6.1	Basisarchitektur . . . . .	358
17.6.2	Integration eines eigenständigen Auskunfts- und Beratungssystems . . . . .	363
17.6.3	Exkurs: Wissenbasierte Systeme und Expertensysteme . . . . .	364
17.7	Resümee . . . . .	372
	Glossar . . . . .	375
	Kurzbiographien . . . . .	383

**XVIII Inhalt**

<b>Personenregister</b> . . . . .	<b>387</b>
<b>Sachregister</b> . . . . .	<b>393</b>

---

# **Einführung**

---



# 1 Fragestellungen und Forschungsstrategien der Software-Ergonomie

Norbert A. Streitz

## 1.1 Ausgangspunkt

In diesem einleitenden Kapitel soll erläutert werden, was **Software-Ergonomie** ist, warum es dieses Forschungsfeld überhaupt gibt und warum es verstärkt betrieben werden sollte. Damit verbunden ist ein kurzer Abriss der historischen Entwicklung von Software-Ergonomie. Im Anschluß daran werden die an diesem inter- und multidisziplinären Arbeitsgebiet beteiligten Disziplinen mit ihren unterschiedlichen Sichtweisen und daraus resultierenden Forschungsstrategien vorgestellt.

*Was ist Software-Ergonomie ?* In Kurzform kann man sagen, daß sich die Software-Ergonomie auf der Basis des Grundverständnisses von Ergonomie, i. e. der Anpassung von technischen Systemen an den Menschen und nicht umgekehrt (!), mit dem besonderen Verhältnis von Softwaresystemen und ihren Benutzern, sowie der Berücksichtigung dieses Verhältnisses bei der Analyse, Gestaltung und Bewertung interaktiver Computersysteme beschäftigt. Diese Charakteristika klingen auch in anderen Namensgebungen an, die in diesem Gebiet Verwendung finden: **Mensch-Computer-Interaktion** (MCI), bzw. im englischen Sprachraum **Human-Computer-Interaction** oder Human Factors in Computing. Dabei kennzeichnet der Begriff Mensch-Computer-Interaktion einen umfassenderen Themenkreis als Software-Ergonomie. So sind Fragen der Gestaltung von Tastaturen, von blend- und flimmerfreien, hochauflösenden Bildschirmen, etc. – also die sog. **Hardware-Ergonomie** – Teil der MCI-Forschung, aber nicht der Software-Ergonomie. Die Software-Ergonomie thematisiert diejenigen Aspekte der MCI, die sich aus der Gestaltung von Software ergeben. Dieser Themenbogen kann sich sehr weit spannen – z.B. von der Gestaltung von Kommandosprachen, Menühierarchien und Fenstersystemen über Ursachen für Beanspruchung und Belastung bis hin zur Funktionsaufteilung zwischen Mensch und Computer. In letzter Zeit findet dabei insbesondere das Verhältnis von benutzerfreundlicher Schnittstellengestaltung und bereitgestellter Funktionalität verstärkte Beachtung. Die dabei erzielten Lösungen haben unmittelbare Rückwirkungen auf die Arbeitsgestaltung. /Hacker 87/ formuliert diese Beziehung, indem er herausstellt, daß Softwaregestaltung als Arbeitsgestaltung zu begreifen ist. Daher werden sich Fragen der Software-Ergonomie nicht auf die unverzichtbare Schnittstellengestaltung beschränken lassen, sondern müssen zusätzlich die Gestaltung der unterstützten Arbeitstätigkeiten einschließen.

Obwohl die zuletzt gemachten Ausführungen fast schon eine Antwort darstellen, wollen wir doch die Frage nach dem Ausgangspunkt stellen und beantworten: „*Warum gibt es das Forschungsfeld Software-Ergonomie und Mensch-Computer-Interaktion ?*“

Orientiert man sich allein an der Zahl der z. Z. in der Bundesrepublik existierenden Bildschirmarbeitsplätze (ca. 2 Millionen, in den USA ca. 35 Millionen Computerterminals), so muß das überaus starke Interesse an diesem Thema auf tieferliegenden Ursachen als die zahlenmäßige Verbreitung dieses Arbeitsmittels zurückzuführen sein. Schließlich gibt es kein eigenständiges Forschungsgebiet „Mensch-Fahrrad-Interaktion“, obwohl es alleine in der Bundesrepublik ca. 35 Millionen Fahrräder und ca. 100 Millionen in den USA gibt. Es gibt 29,3 Millionen Telefonapparate in der BRD, aber keine spezielle Tagung zur „Mensch-Telefon-Interaktion“. Mit Sicherheit gibt es auch mehr Radio- und Fernsehgeräte als Computersysteme, aber keine wissenschaftliche Zeitschrift zum speziellen Thema „Mensch-Radio-Interaktion“. Bei den genannten Beispielen liegt die Betonung immer auf der „Interaktion“, denn natürlich gibt es z. B. in der Verkehrspsychologie Untersuchungen zum Fahrverhalten und zur Verkehrssicherheit. Es gibt ergonomische Überlegungen wie Telefontaster geformt und die Zifferntasten gruppiert sein sollten. Manche Hersteller machen sich sicherlich auch Gedanken darüber, wie und wo Knöpfe und Anzeigen auf Radio- und Fernsehgeräten angeordnet sein sollten – oft mit Folgen, die nicht unter das Stichwort Verständlichkeit und leichte Bedienbarkeit einzuordnen sind. Worauf es uns bei der Diskussion über die Mensch-Computer-Interaktion ankommt, ist der spezielle Charakter der hier vorliegenden Interaktivität, die bei fortgeschrittenen Computersystemen auf das Verhalten des Benutzers reagierende dynamische Reaktionen einschließt. Die Reaktionen der Softwaresysteme sind zwar algorithmisch festgelegt und damit im Prinzip auch vorhersehbar, aber die Fülle an Funktionen und Situationen führen zu einer Komplexität, die einer Mensch-Mensch-Interaktion durchaus vergleichbar sein können. Während die Benutzung eines Fahrrades eine rein mechanische Interaktion darstellt, ein (herkömmliches) Telefon oder Radio eine ziemlich begrenzte Anzahl von Optionen zur Auswahl anbietet, kann bereits ein Anwendungsprogramm mit mittlerem Funktionsumfang in seinen Reaktionen für einen normalen Benutzer nicht in jedem Moment vorhersehbar und damit problemlos bedienbar sein. In dem Maße, in dem bei den zuvor als „einfache“ Systeme geschilderten Geräten zusätzliche Funktionalität über Computer realisiert oder Funktionen vom Menschen interaktiv gesteuert werden (Bordcomputer im Auto, Telekommunikation, etc.), bekommen auch bei diesen Systemen Fragen der Software-Ergonomie eine Bedeutung. Entscheidender Punkt unserer Argumentation ist der Einsatz von Computern und damit die zentrale Rolle interaktiver Software, die anteilig Aufgaben übernimmt, die zuvor vom Menschen ausgeführt wurden.

Wie groß der Anteil computerunterstützter Arbeit auch immer in der Zukunft<sup>1</sup> sein wird, Informationssysteme- und Kommunikationstechnologien werden unser zukünftiges berufliches und privates Leben entscheidend beeinflussen. Es ist aber nicht möglich und auch nicht wünschenswert, die Gesamtheit der Vorgänge im Büro-, Verwaltungs-, Konstruktions- und Produktionsbereich vollständig zu automatisieren. Damit verbleiben immer Anteile der Aufgabenbearbeitung beim Men-

<sup>1</sup> Nach /Giuliano 82/ wird geschätzt, daß im Jahre 1990 zwischen 40% und 50% aller amerikanischen Arbeiter „will use some type of electronic terminal equipment on a daily basis“. Nach /Fährlich 87/ „werden im Jahre 2000 fast 90% aller Berufstätigen mit Computertechnologien am Arbeitsplatz konfrontiert sein“.

schen, die die eingesetzten Systeme auch bedienen und kontrollieren müssen. Mit der zunehmenden Verbreitung von interaktiven Computersystemen in allen Bereichen nimmt dabei die Anzahl derjenigen Benutzer zu, die nicht über eine spezielle Ausbildung im Umgang mit Computersystemen verfügen. Dazu kommt die zunehmende Einsicht, daß interaktive Computersysteme so komplex werden können, daß man sich nicht mehr auf die konventionellen Verfahren von Personalauswahl und Training stützen kann, um eine einfache und fehlerfreie Bedienung und Wartung zu gewährleisten.

Vor diesem Hintergrund ist die Ausgangsfrage nach dem „warum“ einfach zu beantworten. Die Antwort basiert auf der trivialen, aber entscheidenden Beobachtung, daß Computersysteme und Menschen – hier in der Rolle als Benutzer dieser Systeme – sich in wesentlichen Punkten voneinander unterscheiden. Beide Systeme verarbeiten zwar Informationen, z. T. auch die gleichen Informationen, aber auf der Basis unterschiedlicher Wissensbasen und insgesamt auf sehr verschiedene Art und Weise. Diesem Umstand wird aber bei der Gestaltung der Mensch-Computer-Interaktion bei den meisten existierenden Computersystemen nicht oder nur ungenügend Rechnung getragen. Offenbar ist die Mehrzahl der Systementwerfer und Softwareentwickler unfähig, sich vorzustellen oder zu klären, wer ihre Systeme tatsächlich verwenden wird und wozu. Dieses Unvermögen basiert z. T. auf unzureichenden Kenntnissen über Arbeitsabläufe und Aufgabengestaltung sowie über Prinzipien der menschlichen Informationsverarbeitung für die Bereiche Wahrnehmung und Gedächtnis, Denken und Handeln. Diese Defizite führen zu Softwaresystemen, die zu wenig Rücksicht nehmen auf die zu unterstützenden Arbeitsinhalte und auf die Bedürfnisse und Voraussetzungen der sie benutzenden Menschen.

Shneiderman formuliert seinen Ausgangspunkt für die Notwendigkeit von Forschung und praktischer Umsetzung auf dem Gebiet der Mensch-Computer-Interaktion sehr drastisch, wenn er schreibt:

„Frustration and anxiety are a part of daily life for many computer users of computerized information systems. They struggle to learn command language or menu selection systems that are supposed to help them do their job. Some people encounter such serious cases of computer shock, terminal terror, or network neurosis that they avoid using computerized systems. These electronic-age maladies are growing more common; but help is on the way!“ /Shneiderman 87/

Verbunden mit dem Ausmaß der Funktionalität und Komplexität von Anwendungssoftware ist deshalb nur zu oft eine unvollständige, nicht effektive oder sogar fehlerhafte Nutzung zu beobachten. Durch die weitreichenden sozialen und ökonomischen Folgen des Einsatzes von Softwaresystemen kommt der Forschungsthematik *Software-Ergonomie / Mensch-Computer-Interaktion* daher über die Aspekte der individuellen Arbeitssituation hinaus eine globale Bedeutung zu.

Wie äußert sich nun das Interesse an dieser erst seit einigen Jahren verstärkt Beachtung findenden Thematik? Das Interesse wird in einem Anstieg von Verweisen auf diesen Forschungszweig in den Medien und in der allgemeinen Literatur über Computersysteme deutlich, dem Anwachsen von Teilnehmerzahlen auf speziellen Tagungen zur Software-Ergonomie, von entsprechenden thematischen Fachgesprächen auf großen allgemeineren Tagungen und einer Vielzahl von Artikeln in Fach-

zeitschriften und Büchern. Es ist festzustellen, daß sich software-ergonomische Fragestellungen entweder als Teil in den unterschiedlichsten Wissenschaftsdisziplinen oder auch in zunehmenden Maße als eigenständige Disziplin etabliert haben. Daß dies nicht immer so war und wie sich dieses Forschungsgebiet zu dem entwickelt hat, wie es sich heute darstellt, darauf wird in dem Abschnitt über die historische Entwicklung näher eingegangen.

Auch wenn wir die zunehmende Bedeutung der MCI-Forschung hier herausstellen, so ist die von /Shneiderman 87/ gemachte Aussage

„Human engineering, which was seen as the paint put on at the end of a project, is now understood to be the steel frame on which the structure is built.“

sicherlich noch keine Realität. Eine in diesem Sinne verstandene Software-Ergonomie ist noch auf dem Wege, diese Position zu erobern. Dies wird umso mehr der Fall sein, je mehr Computersysteme, wie z. B. Expertensysteme, ihre Anwendungsfelder in bisher der menschlichen Intelligenz und Kreativität vorbehaltenen Bereichen finden. Vor diesem Hintergrund finden dann auch Begriffe wie **kognitive Ergonomie / cognitive ergonomics** /Dzida 80/, /Shackel 81/, /Streitz 86/ oder **cognitive engineering** /Norman 86/, /Rasmussen 87/ ihre Berechtigung. Damit wird deutlich gemacht, daß zunehmend die Computerunterstützung von Arbeitstätigkeiten, die über Routineanforderungen hinausgehen und kognitiv anspruchsvolle Problemlösefähigkeiten erfordern, zum Gegenstand der Forschung wird.

## 1.2 Sachproblem und Interaktionsproblem

Nachdem wir den Ausgangspunkt für software-ergonomische Forschungsthemen identifiziert haben, wollen wir nun den Gegenstandsbereich näher spezifizieren und beispielhaft beschreiben. Bei der Diskussion der Gestaltung interaktiver Systeme werden häufig Forderungen nach Benutzerfreundlichkeit / Benutzbarkeit (*usability*, vgl. /Shackel 85/) und nach Nützlichkeit (*utility*) erhoben. Obwohl diese beiden Anforderungen immer gemeinsam betrachtet werden sollten, kennzeichnen sie doch unterschiedliche Aspekte der Gestaltung der Mensch-Computer-Interaktion. Beschreibt man das Bearbeiten von Arbeitsaufgaben mit interaktiven Anwendungsprogrammen innerhalb des Problemlöseparadigmas, so kann diese Arbeitstätigkeit als *interaktives Problemlösen* konzeptualisiert werden /Streitz 85/. Die Verwendung von interaktiven Computersystemen als Arbeitsmittel führt zu einer Aufteilung der Arbeitsaktivitäten in die Arbeit an der inhaltlichen Problemstellung einerseits und die Benutzung der Arbeitsmittel andererseits<sup>2</sup>. Diese Aufteilung kann durch die Unterscheidung in **Sachproblem** und **Interaktionsproblem** gekennzeichnet werden /Streitz 85, 86/.

<sup>2</sup> Hier ist anzumerken, daß es ein solches Interaktionsproblem im Prinzip in jeder Arbeitssituation gibt, die aber bei interaktiven Computersystemen verstärkt sichtbar wird. So kann z. B. auch die Handhabung von traditionellen Werkzeugen am Anfang ein Problem darstellen.

Diesen Problemtypen entsprechen unterschiedliche Rollen, die z. B. ein Sachbearbeiter übernehmen muß, wenn er seine Arbeitsaufgabe mit Hilfe eines Computersystems bearbeiten möchte. Dabei versteht man unter dem Sachproblem die eigentliche Arbeitsanforderung, im Falle eines Ingenieurs also z. B. die Konstruktion eines Werkstücks, die auf den Vorgaben des Auftraggebers an den Sachbearbeiter basiert. Die Tätigkeit ist dem Konstrukteur prinzipiell zwar vertraut, erfordert aber insbesondere bei neuen Fragestellungen Problemlösefähigkeiten zu bezeichnen. Benutzt er dabei einen Computer (z. B. ein CAD-System) als Arbeitsmittel (oder Werkzeug), muß er mit diesem interagieren. Das Bearbeiten des Sachproblems mit Hilfe des Computersystems erzeugt nun – insbesondere für neue oder gelegentliche Benutzer und solche, die keine spezielle DV-Ausbildung haben – ein neues, zusätzliches Problem: das Interaktionsproblem. Diese Auffassung der Mensch-Computer-Interaktion ist in Abb. 1-1 dargestellt.

Diese Darstellung soll veranschaulichen, daß das Sachproblem nicht auf direktem Wege mit dem im Computer (über den Realitätsbereich) repräsentierten Wissen bearbeitet werden kann, sondern erst nach Überwindung des Interaktionsproblems. Dazu muß der Sachbearbeiter neben der Rolle des (Sach)problemlösers auch die Rolle eines Benutzers (des interaktiven Computersystems) übernehmen. Innerhalb des von uns zur Beschreibung verwendeten Problemlöseansatzes bedeutet dies, daß der Sachbearbeiter entsprechende Wissensrepräsentationen für beide Probleme aufbauen und verwenden muß.

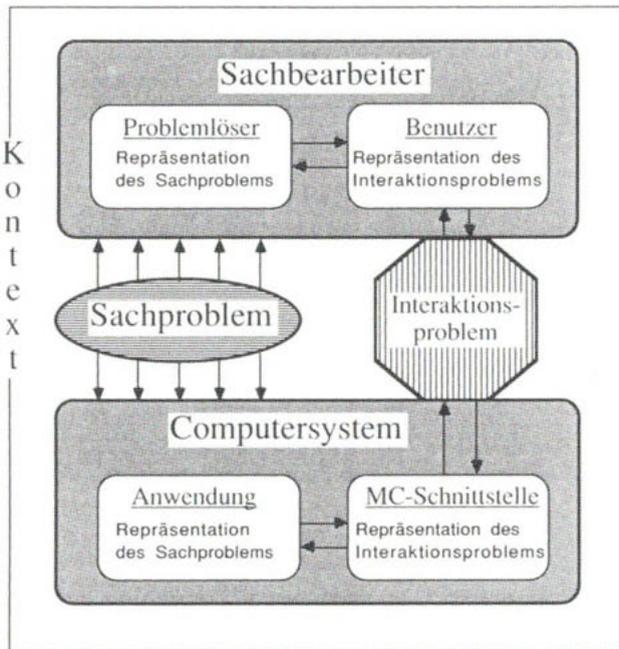


Abb. 1-1: Mensch-Computer-Interaktion als interaktives Problemlösen

Diese unterschiedlichen Aspekte interaktiven Problemlösens spiegeln sich auch in Äußerungen von Personen wider, die existierende Software verwenden. So weist die Bemerkung

„Wenn ich in das Formular auf dem Bildschirm Informationen eintragen will, muß ich zuerst verschiedene Tasten betätigen, um an die gewünschten Stellen zu gelangen.“

auf die Existenz des Interaktionsproblems hin. Während beim Eintragen von Zahlen mit dem Bleistift in ein Papierformular die entsprechenden Bewegungen der Hand unmittelbar mit der Funktion *Schreiben* verbunden sind, erfordert die Positionierung des Cursors (Schreibmarke) auf dem Bildschirm eine zusätzliche Handlungsfolge. Daß die Positionierungsproblematik keine triviale Angelegenheit ist, zeigt die Vielzahl von technischen Lösungsansätzen (z. B. Kontrollkommandos, Cursortasten, Maus, Lichtgriffel, berührungsfähige Folien, Knie-, Fuß- und Kopfsteuerung) (siehe /Balzert, in diesem Band/) und die Fülle von empirischen Untersuchungen zum Vergleich dieser Positionierungshilfsmittel. Die Verwendung von Computerprogrammen zur Bearbeitung von Aufgaben kann aber auch Schwierigkeiten für den inhaltlichen Anteil der Tätigkeit – das Sachproblem – zur Folge haben. So beschreibt die Äußerung

„Ich kann die Aufgaben nicht so bearbeiten, wie ich es zuvor gewohnt war, da die Reihenfolge der Schritte anders ist und auch neue Schritte hinzugekommen sind.“

eine häufige Folge des Einsatzes von Software. Mehr oder weniger inhaltlich-sachlogisch gerechtfertigt hat die Umsetzung von existierenden Arbeitsabläufen in Interaktionen mit einem Anwendungsprogramm die eben beschriebenen Auswirkungen. Dafür können unterschiedliche Gründe wirksam sein. Es führt aber in jedem Fall dazu, daß in unserem Beispiel der Konstrukteur Entscheidungen über die Gestaltung von Details des Werkstücks in einer anderen Reihenfolge treffen muß als zuvor üblich. Damit wird sein Problemlöseverhalten geändert, u. U. in entscheidender Weise. Es erfordert nicht nur ein Umdenken auf seiten des Konstrukteurs, sondern kann auch Rückwirkungen auf das Ergebnis der Konstruktionstätigkeit haben, da die meisten Konstruktionsentscheidungen nicht unabhängig voneinander sind. Dies ist auch ein Beispiel für den Einfluß von unterschiedlichen Beschreibungswelten, die zur Vermittlung der Problemstellung verwendet werden. Unterschiedliche Beschreibungs-/Metaphernwelten für Arbeitsinhalte und für die Interaktionsmöglichkeiten (z. B. Kommando- und Menübezeichnungen) setzen unterschiedliche Wissensrepräsentationen voraus und induzieren mentale Modelle /Streitz 88a/. Die Verfügbarkeit und Adäquatheit der mentalen Modelle wirken sich wiederum entscheidend auf den Problemlöseprozeß aus (für eine Übersicht siehe /Streitz 88b/).

Obwohl diese Beispiele für das Auftreten des Interaktions- und des Sachproblems zunächst unverbunden erscheinen, sollten Gestaltungsmaßnahmen in bezug auf diese beiden Aspekte nicht unabhängig voneinander durchgeführt werden. So hat z. B. die Verfügbarkeit bestimmter Positionierungsinstrumente (Interaktionsproblem) Rückwirkungen auf mögliche grafische Darstellungsformen, auf die Art der entstehenden Konstruktionszeichnungen und damit auf die Bearbeitung des Sachproblems.

Hier wird deutlich, daß die Bewertung eines Computersystems bzw. einer bestimmten Software in bezug auf ihre Verwendbarkeit zur Erledigung von Arbeitsaufgaben immer sowohl über die Gestaltung der Benutzerschnittstelle (Interaktionsproblem) als auch über die Funktionalität (Sachproblem) erfolgen muß.

Um einen Eindruck von der Bandbreite der Funktionalität zu geben, die in unterschiedlichen Anwendungen erforderlich ist, geben wir nachfolgend – ohne Anspruch auf Vollständigkeit – einige Beispiele für Klassen von Benutzergruppen und Situationen. So sind Mensch-Computer-Schnittstellen zu gestalten für:

- Situationen mit hoher mentaler Belastung und psychischem Druck (Fluglotsen, Piloten, Prozeßkontrolle)
- Situationen mit hohen Sicherheitsanforderungen (Versicherungs- und Bankbereich, Kernkraftwerke, militärische Anwendungen)
- behinderte Personen (blind, taub, motorisch behindert)
- Situationen und Personen, bei denen keine Voraussetzungen über Wissensstand / Training gemacht werden können (öffentliche Auskunftssysteme)
- Personen mit unterschiedlichen Sprachen und kultureller Herkunft (bei der Flugreservierung, Auskunftssysteme auf Bahn- und Flughäfen)
- das Herstellen von Transparenz und Selbsterklärungsfähigkeit von Systemen (für Wartungs-, Diagnose- und Reparaturmaßnahmen)
- das Navigieren und Suchen in sehr großen Datenbanken (Fachinformationssysteme, Patentwesen, Wirtschaftsdatenbanken)
- die Unterstützung von kooperativem Arbeiten mit räumlich / zeitlich verteilten Teilnehmern (Entscheidungssysteme für Manager, Videokonferenzen, große Software-Entwicklungsteams, Autoren- und Publikationsteams)

## 1.3 Forschungsziele und Systementwurf

Die aufgeführten Beispiele machen deutlich, daß Forschung zur Software-Ergonomie nicht als Selbstzweck betrieben wird. Obwohl aufzuteilen auf die drei Hauptgebiete – Analyse, Gestaltung und Evaluation – gibt es eine eindeutige Zielsetzung: die Entwicklung interaktiver Computersysteme zu ermöglichen, die vorgegebene ergonomische Kriterien erfüllen. Dazu gehören einerseits theoretische und empirische Untersuchungen, um Kriterien aufstellen zu können. Andererseits müssen diese Erkenntnisse als Vorgaben für existierende Systeme umgesetzt werden. Wie diese Kriterien im einzelnen aussehen, kann nicht Gegenstand dieses einleitenden Kapitels sein (vgl. dazu die folgenden Einzelbeiträge). Es können aber einige globale Vorgaben benannt werden, die als Orientierung und als Rahmenbedingungen angesehen werden können.

Die Diskussion der beiden Rollen interaktiver Problemlöser in bezug auf das Sach- und das Interaktionsproblem hat Konsequenzen für den Entwurf interaktiver Systeme. Sie spiegeln sich wider in der *Forderung nach aufgabenzentrierten und benutzerorientierten Systemen*. Die **Aufgabenzentriertheit** erfordert, daß die zur

Bearbeitung des Sachproblems erforderliche Funktionalität in aufgabenkompatibler und flexibler Weise (u. U. also auch mehrere Bearbeitungsverfahren für das gleiche Problem) zur Verfügung gestellt wird. Damit sollen die Voraussetzungen zur optimalen Verwendbarkeit in Hinblick auf die Nützlichkeit (*utility*) des Systems geschaffen werden.

**Benutzerorientiertheit** bedeutet, daß durch Maßnahmen zur Minimierung des Interaktionsproblems das Ausmaß an Bedienbarkeit (*usability*) entscheidend erhöht wird. Dazu ist die These der *kognitiven Kompatibilität* aufgestellt worden, die ein Minimum an Diskrepanzen zwischen Vorwissenstrukturen beim Benutzer einerseits und in dem Computersystem verwendeten Wissensrepräsentationen andererseits gewährleisten soll /Streitz 87/. Die entsprechenden Maßnahmen basieren auf dem Grundverständnis von Ergonomie, daß die Technik auf die Fähigkeiten und Möglichkeiten, aber auch auf die Grenzen menschlicher Informationsverarbeitung ausgerichtet werden muß. Liegen entsprechende wissenschaftlich abgesicherte Befunde über spezielle Eigenschaften des Menschen noch nicht vor, dann müssen diese Voraussetzungen für Gestaltungsmaßnahmen selbst erforscht werden. Hieran zeigt sich die enge Verbindung und gegenseitige Anregung von Grundlagenforschung und angewandter Forschung innerhalb der Software-Ergonomie.

In einem umfassenderen Selbstverständnis von Software-Ergonomie werden ergänzende und weitergehende Betrachtungen und Anforderungen, wie sie in den Arbeitswissenschaften ganz allgemein für Arbeitssituationen aufgestellt werden (z. B. zur Arbeitszufriedenheit, zur Persönlichkeitsförderlichkeit und zur organisationalen Einbettung von Arbeit), einbezogen und für den hier vorliegenden Gegenstandsbereich präzisiert (siehe z. B. /Spinas, Troy, Ulich 83/, /Balzert 86/, /Hacker, Schönfelder 86/, /Skarpelis 87/, /Ulich, in diesem Band/). Wir erinnern in diesem Zusammenhang auch noch einmal an unsere bereits zu Beginn gemachte Bemerkung, daß Software-Gestaltung zugleich Arbeitsgestaltung ist.

Wie die genannten Zielsetzungen erreicht werden können, darüber entscheiden die gewählten Forschungs- und Entwicklungsstrategien, die in Abschnitt 1.6 behandelt werden.

## 1.4 Hardware- und Software-Entwicklung im Rückblick

Im Rahmen einer Einführung in die Software-Ergonomie erscheint es angebracht, auch einige Anmerkungen zur historischen Entwicklung und Einordnung dieses Gebietes zu machen. Dabei kann es hier nicht um Vollständigkeit und intensives Quellenstudium gehen. Dazu sei z. B. auf /Shackel85/ verwiesen oder auf /Gaines 86/, der die Entwicklung „*from timesharing to the sixth generation*“ nachgezeichnet hat. Dabei wird die Entwicklung der MCI-Forschung im Rahmen und im Vergleich zur allgemeinen Hardware- / Software-Entwicklung und zur Forschung auf dem Gebiet der Künstlichen Intelligenz betrachtet (vgl. Tab. 1-1).

Zu Beginn des Computerzeitalters, d. h. zwischen 1948 und 1955, mußten die Benutzer von Computern erst selbst zu Computerspezialisten werden. Dies erforderte viel Zeit und Aufwand, aber es gab genügend Personen, die dazu bereit waren, weil

Tab. 1-1: Entwicklungen bei Rechnern, Künstlicher Intelligenz und Mensch-Computer-Interaktion über sieben Generationen /Gaines 86/

	<b>Hardware/Software EDT/VMA/POL</b>	<b>State of AI KBS/IIS/AAS</b>	<b>State of HCI HCI</b>
<b>0</b> 1940 - 47	<b>Up and down</b> Relays to vacuum tubes COLOSSUS, ENIAC	<b>Mind as Mechanism</b> Logic of neural networks <i>Behavior, purpose &amp; teleology</i>	<b>Designer as User</b> Judge by ease of use
<b>1</b> 1948 - 55	<b>Gee Whiz</b> Tubes, delay lines, drums BINAC, EDSAC, UNIVAC, WHIRLWIND, EDVAC, ACE IBM 701, 702, 650 Numeric control, nav aids	<b>Cybernetics</b> Turing test Ashby's homeostat Grey Walter's tortoise Samuel's checkers player <i>Design for a Brain</i>	<b>Machine Dominates</b> Person adapts to machine <i>Human Use of Human Beings</i>
<b>2</b> 1956 - 63	<b>Paper Pushers</b> Transistors & core stores I/O control programs IBM 704, 7090, 1401, NCR 315 UNIVAC 1103, PDP 1, 3, 4, 5 FORTRAN, ALGOL, COBOL Batch, execs, supervisors Petrinets <i>Communications of ACM</i>	<b>Generality/Simplicity The Oversell</b> Perceptron, GPS, EPAM Learning machines Self-Organizing systems IPL V, LISP 1.5 Dartmouth AI Conference <i>Mechanization of Thought Processes</i>	<b>Ergonomics</b> Console ergonomics Job control languages Simulators, graphics CTSS, MAC JOSS, BASIC  <i>Breakthrough to HCI</i>
<b>3</b> 1964 - 71	<b>Communicators</b> Large-scale ic's Interactive terminals IBM 360, 370 CDC 6600, 7600 PDP 6, 7, 8, 9, 10 DBMS, relational model Intel 1103, 4004	<b>Perform by Any Means</b> Semantic nets, ATNs, ELIZA PLANNER, fuzzy sets DENDRAL, scene analysis Resolution principle, 1st ICAI <i>Machine Intelligence I Artificial Intelligence Breakthrough to KBS</i>	<b>Man-Machine Studies Interactive Experience</b> Time-sharing services Interactive terminals Speech synthesis TSS360, APL360 Unix, shell <i>Int. J. Man-Machine Studies</i>
<b>4</b> 1972 - 79	<b>Personal Resources</b> Personal computers Supercomputers VLSI Very large file stores Databanks, videotex IBM 370/168 - MOS memory & virtual memory Intel 8080, NS PACE 16-bit	<b>Encoded Expertise &amp; Over-Reaction</b> PROLOG, Smalltalk, frames Scripts, systemic grammars SHRDLU, MARGIE MYCIN, TEIRESIAS Dreyfus, Lighthill & Weizenbaum attacks on AI <i>Cognitive Science</i>	<b>HCI Design Rules</b> Personal computing Dialog rules LUNAR, ROBOT, LIFER Videotex services Altair & Apple PC's Visicalc <i>J. Assn. Comput. Linguistics Byte</i>
<b>5</b> 1980 - 87	<b>Action Aids</b> PC's with power & storage of mainframes plus graphics & speech processing Networks, utilities OSI, NAPLS standards IBM 370 chip, HP-9000 chip with 450.000 transistors	<b>Commercialization</b> LISP and PROLOG machines Expert system shells EMYCIN, AL/X, OPSS, APES Fifth generation project ICOT PSIM & PIM Knowledge bases <i>Handbook of AI</i>	<b>User-Natural Systemic Principles</b> Xerox Star, IBM PC Apple Macintosh Videodisk Human protocol KLONE-ED, INDIS, KLAUS ARGOT, HAM-ANS
<b>6</b> 1988 - 93	<b>Partners</b> Optical logic and storage Organic processor elements AI in routine use	<b>Learning &amp; Emotion</b> Parallel knowledge systems Audio and visual sensors Multi-modal modeling	<b>User-Similar Automated Design</b> Integrated multi-modal systems Emotion detection

sie die Möglichkeiten und den potentiellen Nutzen erkannten. In dieser Zeit spielten ergonomische Überlegungen kaum eine Rolle (auch wenn /Gaines 84, 86/ nicht müde wird, immer wieder Mauchly mit einer Bemerkung aus dem Jahre 1947 zu zitieren, in dem erste ergonomische Überlegungen anklingen.) Die Verfügbarkeit einer bestimmten Funktionalität, wie z. B. ein bestimmtes Berechnungsverfahren, stand im Vordergrund. Die Situation begann sich Ende der 50er bis Anfang der 60er Jahre zu wandeln. Die ersten ergonomischen Arbeiten, die sich mit Computersystemen beschäftigten, wurden sporadisch durchgeführt, ohne theoretische Grundlage und waren hauptsächlich an Hardware-Fragen orientiert. Diese Arbeiten waren verstreut, wurden – mit einigen Ausnahmen wie z. B. /Shackel 59/ – meistens in den USA durchgeführt und bezogen sich vornehmlich auf militärische Anwendungen.<sup>3</sup>

Die Entwicklung von Minicomputern und von *time-sharing*-Systemen mit verteilten Terminals, die interaktives Arbeiten zu erlauben begannen, hatte Ende der 60er und Anfang der 70er Jahre eine erste Ausweitung der Gruppe möglicher Benutzer zur Folge. Es waren nicht mehr notwendigerweise nur Computerspezialisten. Zu dieser Zeit wurden ergonomische Probleme bei der Computerbenutzung auch in wissenschaftlichen Veröffentlichungen angesprochen /Nickerson 69/, /Shackel 69/. Aber erst das Auftauchen des Mikrocomputers und seine weite Verbreitung in den 80er Jahren zusammen mit seiner Propagierung zum Personal Computer veranlaßte die Computerindustrie, den Aspekt der Benutzerfreundlichkeit als Problem und als Marketingfaktor zugleich anzuerkennen. Parallel dazu wurde die Palette der Computeranwendungen immer breiter und folglich auch die Gruppe der Benutzer immer heterogener, was Ausbildungsstand und Anwendungsanforderungen betrifft. Heutzutage kommt eigentlich jeder als potentieller Benutzer in Frage, insbesondere dann, wenn man an die Integration der eigentlichen Computersysteme mit den Kommunikationssystemen zu integrierten und umfassenden Informationssystemen denkt. Wenn die Qualität der Mensch-Computer-Schnittstelle (inkl. Handbücher etc.) nicht den Anforderungen der Benutzer entspricht, d. h. die Interaktion mit dem System zu schwierig zu erlernen ist, dann müssen die Hersteller solcher Systeme damit rechnen, daß ihre Produkte nicht gekauft werden. Aber von der Einsicht in die Notwendigkeit software-ergonomischer Überlegungen bis zur Umsetzung bei der Produktentwicklung ist es immer noch ein weiter Weg. Andererseits sind die Möglichkeiten, die Schnittstelle mit zusätzlichen Komponenten zu versehen, durch die dramatische Reduzierung der Kosten für Hardware (z. B. für Speicherkapazität) und die Steigerung der Rechengeschwindigkeit erheblich gestiegen. So ist es nicht verwunderlich, daß wir seit Ende der 70er Jahre einen Anstieg der Forschungs- und Entwicklungsaktivitäten auf dem Gebiet der Software-Ergonomie zu verzeichnen haben, die diese Möglichkeiten zu nutzen versuchen.

<sup>3</sup> Die Beschäftigung mit ergonomischen Fragestellungen ist eine Erscheinung, die vor allem in den USA unter der Bezeichnung *human factors engineering* oder auch *engineering psychology* in den 40er Jahren (d. h. während und nach dem 2. Weltkrieg) ihren Ursprung zum großen Teil im militärischen Bereich hatte. So erkannte das Department of Defense (DOD) sehr zeitig die Rolle des „menschlichen Faktors“ für die Sicherheit und Zuverlässigkeit bei der Bedienung der immer komplexer werdenden technischen Systeme.

Ab Anfang bis Mitte der 80er Jahre tauchen dann die ersten Früchte dieser Aktivitäten in Form von kommerziellen Systemen auf, wie z. B. der Xerox Star, Apple Lisa und Macintosh. Die innovativen Interaktionsmöglichkeiten werden realisiert mit Hilfe von hochauflösenden Grafikbildschirmen, Fenstersystemen, Zeigeinstrumenten für die direkte Manipulation, Metaphernwelten, die visuell über Piktogramme dargestellt werden, wie die Schreibtischmetapher. Heute gehören diese Merkmale zum Standard bei fortgeschrittenen Arbeitsplatzrechnern (*workstations*) wie sie in vielen Forschungsinstituten verwendet werden. Gemessen an der Vielzahl der insgesamt existierenden Bildschirmarbeitsplätze sind diese Hardware-Voraussetzungen aber noch nicht allgemeiner Standard.

## 1.5 Die Wissenschaftslandschaft

Veränderungen in der wissenschaftlichen Landschaft können in jedem Fachgebiet an dem Erscheinen einflußreicher Bücher, dem Abhalten bedeutender Tagungen und dem Gründen von themenspezifischen Fachzeitschriften sowie wissenschaftlichen Gesellschaften abgelesen werden. So auch bei der Software-Ergonomie.

Nach /Gaines 86/ wird das 1965 in Yorktown Heights durchgeführte IBM *Scientific Computing Symposium on Man-Machine Communication* als die erste wissenschaftliche Tagung zu diesem Thema angesehen. 1969 erschien die erste Nummer des *International Journal of Man-Machine Studies*. Anschließend erschienen die ersten Bücher zu diesem Thema z. B. *Man-Computer Problem Solving* /Sackman 70/, *The Psychology of Computer Programming* /Weinberg 71/, *Design of Man-Computer Dialogues* /Martin 73/. In diese Zeit fällt auch ein Anstieg von Veröffentlichungen zu software-ergonomischen Themen in anderen Zeitschriften, wie *Human Factors*, *Ergonomics*, *Communications of the ACM*. Die Folgezeit sah und sieht eine Vielzahl von Buchveröffentlichungen, von denen hier nur einige exemplarisch genannt werden sollen: /Badre, Albert, Shneiderman 80/, /Shneiderman 80/, /Card, Moran, Newell 83/, /Thomas, Schneider 84/, /Vassiliou 84/, /Hartson 85/, /Norman, Draper 86/, /Shneiderman 87/. Seit 1985 erscheint die Fachzeitschrift *Human-Computer Interaction*.

Seit 1982 wird in den USA jährlich (außer 1984) die sog. CHI-Konferenz als Veranstaltung der *Special Interest Group on Computer-Human Interaction der Association for Computing Machinery* (SIGCHI der ACM) durchgeführt. 1984 findet die First USA-Japan Conference on Human-Computer Interaction in Hawaii statt /Salvendy 84/. Beide Tagungen hatten bereits ihre Folgeveranstaltungen /Salvendy 87/, /Salvendy, Sauter, Hurrell 87/, wobei die zweite INTERACT-Konferenz in Stuttgart stattfand /Bullinger, Shackel 87/. Darüber hinaus gab und gibt es natürlich eine Reihe kleinerer themenspezifischer Tagungen.

Die Tatsache, daß die beiden INTERACT-Konferenzen in Europa stattfanden, liegt einerseits natürlich in der unbestrittenen Rolle von Brian Shackel begründet, ist andererseits aber auch Ausdruck der wachsenden Aktivitäten in Europa. Hier lag der Schwerpunkt zunächst mehr auf der ergonomischen Gestaltung von Bildschirmarbeitsplätzen im Sinne der Hardware-Ergonomie, wie z. B. in den Büchern von

/Grandjean, Vigliani 80/ und /Cakir, Hart, Stewart 80/ dargestellt. Aber auch software-ergonomische Themen mit Berücksichtigung kognitiver Aspekte wurden in Europa auf Tagungen und in Büchern behandelt /Smith, Green 80/, /Coombs, Alty 81/, /Shackel 81/, /Sime, Coombs 83/. 1982 erschien die Fachzeitschrift *Behavior and Information Technology* (BIT). Im europäischen Rahmen entwickelte sich auch eine eigenständige Tagungsreihe, die seit 1982 alle zwei Jahre als *European Conference on Cognitive Ergonomics* (ECCE) insbesondere die kognitiven Aspekte in den Vordergrund stellt /Green, Payne, van der Veer 83/, /van der Veer et al. 84/, /Falzon et al. 88/. Diese Aktivitäten führten dann 1987 zu der formellen Gründung der European Association of Cognitive Ergonomics (EACE) mit Sitz in Paris.

Hauptsächlich durch europäische Forscher getragen, aber international ausgerichtet, ist das MACINTER-Network der *International Union of Psychological Sciences on Man-Computer Interaction Research*. Es wurde 1984 mit einer ersten Tagung /Klix, Wandke 86/ gestartet und hat in der Zwischenzeit eine Reihe von Workshops und Symposia in Zusammenhang mit anderen größeren Veranstaltungen durchgeführt. 1988 wurde wiederum eine Überblickstagung /Klix et al. 88/ veranstaltet.

Aber auch in Deutschland war man nicht untätig. So gibt es seit 1981 die jährlich stattfindenden Arbeitstagungen zur Mensch-Maschine-Kommunikation, in denen ein kleiner Kreis von Wissenschaftlern und Praktikern diskutierte. 1983 fand das Thema in Nürnberg mit der Tagung *Software-Ergonomie* /Balzert 83/ zum ersten Mal ein größeres Forum im deutschsprachigen Raum. Im Anschluß daran wurde im gleichen Jahr innerhalb der Gesellschaft für Informatik (GI) der Fachausschuß *Ergonomie in der Informatik* und die Fachgruppe *Software-Ergonomie* gegründet. Der Erfolg dieser ersten Veranstaltung setzte sich fort in einem zweijährigen Rhythmus mit den Tagungen in Stuttgart /Bullinger 85/ und in Berlin /Schönpflug, Wittstock 87/. Aufgrund des allseits erkannten Defizits an Ausbildungsmöglichkeiten auf diesem interdisziplinären Wissen erfordernden Gebiet wurde 1986 erstmals die Software-Ergonomie Herbstschule (SEH) durchgeführt, die seitdem jährlich stattfindet.

Auf der Basis dieses Überblicks erscheint es gerechtfertigt festzustellen, daß die Software-Ergonomie / Mensch-Computer-Interaktion inzwischen ein auf vielfältigen Ebenen etabliertes Forschungs- und Entwicklungsgebiet ist, das hoffentlich auch in der Zukunft eine Fülle von wichtigen Ergebnissen zu unser aller Nutzen und Freude produzieren wird.

## 1.6 Beteiligte Disziplinen und Forschungsstrategien

Aus den bisher gemachten Ausführungen ist unmittelbar einsichtig, daß die Fragestellungen der Software-Ergonomie verschiedene wissenschaftliche Disziplinen berühren. In diesem Abschnitt wollen wir diese und ihr Verhältnis zueinander näher betrachten.

Computersysteme können zunächst einmal aus der rein technischen Perspektive betrachtet werden. Die *Informatik* stellt eine Reihe von Ansätzen zum Entwurf und zur Realisierung von Software-Systemen bereit. Damit liegt eine Wurzel der Software-Ergonomie in der Informatik. Wie schon in dem historischen Abriss angedeutet, ist die Entwicklung interaktiver Computersysteme bisher weitgehend durch Fortschritte in der Hardware- und Software-Entwicklung bestimmt worden. Die Umsetzung dieser Möglichkeiten geschah aber meistens allein unter dem Gesichtspunkt der technischen Machbarkeit und weniger gemäß den Anforderungen der Benutzer und ihren Aufgabenstellungen.

Die Tatsache, daß diese Systeme von Menschen benutzt werden, macht den Mensch und die zu erledigende Aufgabe gemäß unserem zuvor dargestellten Grundverständnis zum zentralen Ausgangspunkt. Dadurch ist die Bedeutung der *Psychologie* – als der Wissenschaft vom menschlichen Verhalten – für den Systementwurf unmittelbar deutlich. Sie übernimmt für die Einlösung der Forderung nach benutzerorientiertem und aufgabenzentriertem Systementwurf eine entscheidende Rolle, insbesondere wenn es um die kognitiven Aspekte der Mensch-Computer-Interaktion geht.

Der Umgang mit Computern erfolgt meistens im Rahmen von Arbeitsaufgaben und in einer bestimmten Arbeitsumgebung. Über die allgemeinen Prinzipien menschlichen Verhaltens hinaus gibt es damit Abhängigkeiten vom (Arbeits-) Kontext, in dem dieses Verhalten gezeigt wird. Diese Fragen berühren Ansätze aus den *Arbeitswissenschaften* als der Wissenschaft von den Erscheinungsformen der menschlichen Arbeit, von den Voraussetzungen und Bedingungen, unter denen sich Arbeit vollzieht, und den Wirkungen und Folgen, die sie auf Menschen ausübt.<sup>4</sup>

Damit sind die drei beteiligten Hauptdisziplinen identifiziert: Informatik, Psychologie und Arbeitswissenschaften. Dabei ist natürlich zu berücksichtigen, daß nicht immer der gesamte Umfang der jeweiligen Disziplin eine Rolle spielt, sondern immer nur in gewissen relevanten Ausschnitten. Über die unterschiedlichen Perspektiven hinaus ist zu beachten, daß jede Disziplin auch ihre spezifischen Methoden mitbringt und damit verbunden ihre Sichtweise, Forschung zu betreiben, zu bewerten und zu akzeptieren. Forscher, die aus den jeweiligen Disziplinen kommen, haben auch ein unterschiedlich ausgeprägtes Erkenntnisinteresse, wenn sie sich mit Software-Ergonomie beschäftigen. So ist dem einen neben der Verbesserung von konkreten Arbeitssituationen auch an Fortschritten in den Grundlagen seiner Disziplin gelegen. Der Gegenstandsbereich der Software-Ergonomie übernimmt dann die Funktion einer Testumgebung für Grundlagenuntersuchungen, z.B. für die Weiterentwicklung von Theorien über das menschliche Verhalten in komplexen Situationen, wie sie bei der Mensch-Computer-Interaktion auftreten. Ganz allgemein kann man aber festhalten, daß es natürlich Arbeitsteilungen zwischen den Beteiligten geben muß. Diese ergeben sich daraus, ob zu einem bestimmten

<sup>4</sup> Wir wollen uns hier nicht streiten, ob diese Fragen, wie z. B. die organisatorische Einbettung der Arbeitsaufgabe, zu den Arbeitswissenschaften gehören oder nicht eher einen abgrenzbaren Gegenstand der Arbeits- und Organisationspsychologie darstellen. Es gibt da weniger Probleme bei der Zuordnung von Gegenstandsbereichen als vielmehr einen Unterschied in den Methoden, die bei den Arbeitswissenschaftlern eher ingenieurwissenschaftlich orientiert sind.

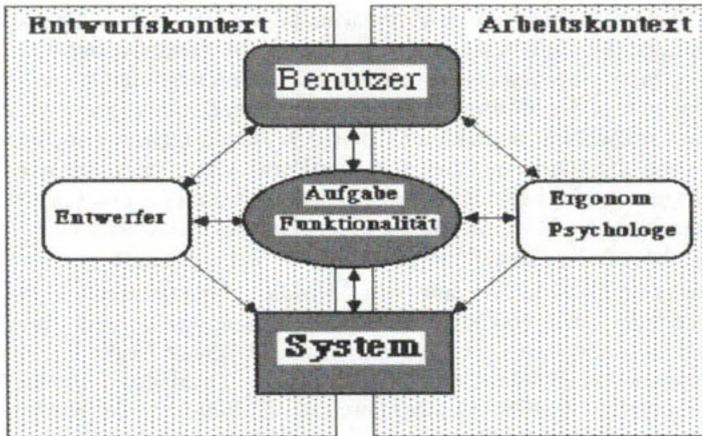


Abb. 1-2: Unterschiedliche Kontexte der Systementwicklung

Zeitpunkt des Entwicklungsprozesses gerade mehr die Analyse der Aufgabe und des Benutzers innerhalb des existierenden und des zukünftigen Arbeitskontextes im Mittelpunkt steht oder ob es sich mehr um die konkreten Überlegungen zur technischen Planung, zur Auswahl von Realisierungskonzepten und deren Umsetzung in Prototypen innerhalb des **Entwurfskontextes** handelt (siehe auch Abb. 1-2).

### Forschungs- und Entwicklungsstrategien

Verbunden mit der Vielfalt der Disziplinen stellt sich die Frage, ob ein *interdisziplinärer* oder *nur ein multidisziplinärer* Ansatz verfolgt werden soll. So vertritt z.B. /Rasmussen 87/ die These, daß nur ein multidisziplinärer Ansatz sinnvoll und erfolgreich zu sein verspricht. Es hängt davon ab, ob die Integration von Wissen aus verschiedenen Disziplinen leichter innerhalb einer Person oder zwischen mehreren Personen zu bewerkstelligen ist. Die Befürchtung, daß der Anspruch, wirklich interdisziplinär zu sein, zu hoch ist, sollte uns aber nicht abhalten, dieses Ziel als Idealvorstellung beizubehalten. Inwieweit es erreicht werden kann unter den v. a. in Deutschland sehr festen Grenzziehungen zwischen den Fakultäten, soll an dieser Stelle nicht diskutiert werden. Auf jeden Fall sollte es aber möglich sein, daß alle am Systementwicklungsprozeß Beteiligten sich soviel Wissen über den jeweils anderen Fachbereich aneignen, daß eine reibungslose Kommunikation in einem Entwicklungsteam gewährleistet ist. Aber auch hier gibt es unterschiedliche Sichtweisen.

So vertreten /Gould, Lewis 85/ den Standpunkt, daß der Entwurf grundsätzlich empirisch orientiert anzugehen ist. Entwerfer sollen sich viel Wissen über die Benutzer und die Arbeitsaufgabe aneignen. Sie sollen das Revidieren von Entwürfen von Anfang an einplanen. Diese Revisionen sollen auf der Basis von empirischen Messungen von Erfolg und Problemen bei tatsächlicher Benutzung z. B. von Prototypen erfolgen.

Ein anderer Standpunkt wird durch den Ansatz von /Card, Moran, Newell 83/ vertreten, der darauf basiert, daß kognitionspsychologische Untersuchungen quantitativ formulierbare Regeln entdecken, die in gewisser Annäherung Vorhersagen für die Gestaltung der Schnittstelle bzw. Komponenten davon machen. Diese können dann von Systementwerfern als Anweisungen verwendet werden.

Eine dritte Vorgehensweise wird oft gar nicht als solche explizit formuliert, ist aber durch das tatsächliche Vorgehen implizit gegeben: Die Ansicht, daß Schnittstellengestaltung vornehmlich technologiegetrieben ist. Verbesserungen resultieren z. B. aus der Verfügbarkeit von größeren Bildschirmen, dem Erfinden neuer Hilfsmittel zur Cursorpositionierung, etc.

In den im Abschnitt 1.3 aufgestellten Zielsetzungen haben wir uns für aufgabenorientierten und benutzerorientierten Systementwurf ausgesprochen. Es stellt sich nun die Frage, wie dieser Anspruch zu erreichen ist. Dazu sollten unserer Meinung nach u. a. die Erkenntnisse der (kognitiven) Psychologie über die menschliche Informationsverarbeitung für die Konstruktion und Gestaltung von Software-Systemen nutzbar gemacht werden. Die Verbindung von Grundlagenforschung und angewandter Systementwicklung bedarf aber einer Forschungsstrategie, die der gegenseitigen Anregung und Förderung dient. Ein Vorschlag dazu /Streitz 87/ besteht in der in Abb. 1-3 skizzierten Vorgehensweise. Bei dieser Darstellung ist zu beachten, daß es sich dabei nicht um eine umfassende Strategie zur Entwicklung von Software-Systemen handelt. Unsere Vorschläge sind durch Methoden des Software-Engineering, die hier aber zunächst einmal ausgeklammert werden sollen, zu ergänzen. Wenn es das Ziel ist, benutzerorientierte Software-Systeme zu erstellen, dann verlangt dies eine allgemeine Theorie menschlichen Verhaltens, die bei diesem Vorschlag der Psychologie entstammt. Aus einer solchen Theorie können Modelle der Mensch-Computer-Interaktion abgeleitet werden, indem als Gegenstandsbereich des menschlichen Verhaltens das Arbeiten mit interaktiven Computersystemen gewählt wird und die allgemeinen Aussagen der Theorien zur menschlichen Informationsverarbeitung darauf bezogen werden. Auf der Basis der dazu formulierten prädiktiven Modelle können dann Vorhersagen abgeleitet werden, die als Arbeitshypothesen wiederum einer Überprüfung unterzogen werden können und auch müssen. Software-ergonomisches Wissen kann aber auch in anderer Form als nicht allgemein formulierte, sondern als bereichsspezifische Theorie oder als praktisches Expertenwissen von Entwerfern und erfahrenen Benutzern vorliegen. Entsprechend kann die Überprüfung auf unterschiedliche Art und Weise erfolgen:

- experimentelle Laboruntersuchungen,
- Felduntersuchungen,
- Fallstudien.

Dabei entsprechen den verschiedenen Formen des vorliegenden Wissens auch bestimmte Überprüfungsverfahren. Experimentelle Laboruntersuchungen dienen hauptsächlich der Überprüfung detaillierter Vorhersagen prädiktiver Modelle. Sie weisen, wenn sie mit einer ausreichenden Anzahl von Versuchspersonen und unter kontrollierten Bedingungen durchgeführt werden, eine hohe interne Validität auf. Allerdings sind sie auch sehr aufwendig, wenn sie die Komplexität der Mensch-Computer-Interaktion einigmaßen vollständig erfassen sollen. (Als ein Beispiel

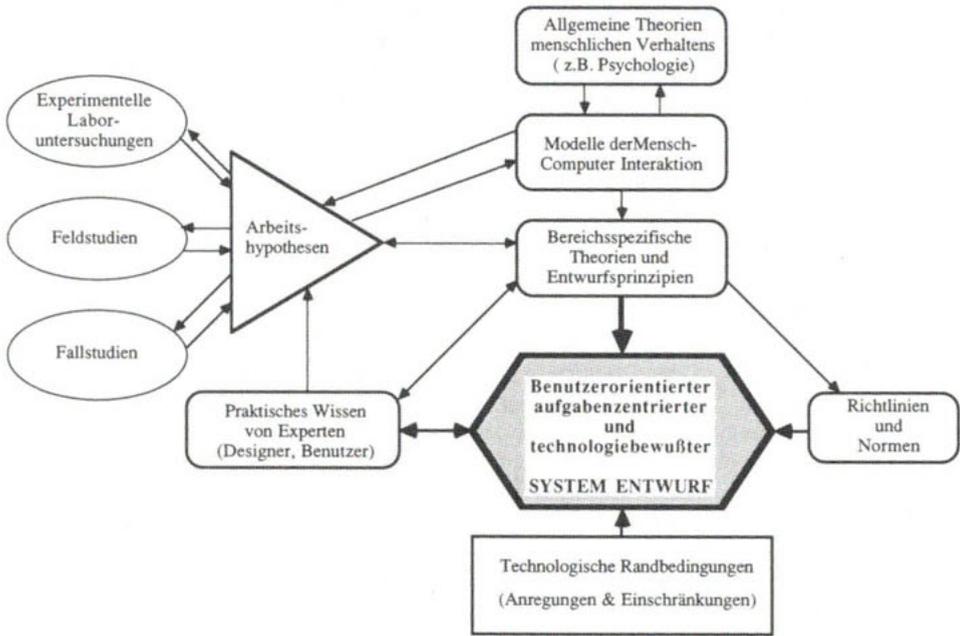


Abb. 1-3: Forschungsstrategien und -methoden im Bereich der Software-Ergonomie

dafür sei hier die Serie von sechs aufwendigen experimentellen Untersuchungen zur mentalen Belastung und kognitiven Prozessen bei komplexen Dialogstrukturen angeführt /Streitz, Eberleh 88/.)

Die Vorhersagen bereichsspezifischer Theorien, die oft auch als Entwurfprinzipien formuliert vorliegen, lassen sich eher in Feldstudien überprüfen, d. h. daß hier zwar die strenge Kontrolle der experimentellen Bedingungen nicht mehr ausreichend gegeben ist, aber die Verallgemeinerbarkeit auf reale Situationen und Aufgaben untersucht werden kann.

Das Praxiswissen der Experten resultiert dagegen meistens aus Fallstudien, die keinen Anspruch auf Verallgemeinerbarkeit erheben können, aber trotzdem ihre Berechtigung innerhalb des Gesamtprozesses der benutzerorientierten Systementwicklung haben, z. B. zur Generierung von Hypothesen. Es ist offensichtlich, daß das Testen der verschiedenen Arbeitshypothesen mit prototypischen Benutzerschnittstellen Teil eines iterativen Prozesses ist, der zu Modifikationen auf den verschiedenen Beschreibungsebenen führt. Im weiteren Verlauf werden sich Teile des akkumulierten Wissens in Richtlinien und Standards (DIN, ISO) niederschlagen und als Randbedingungen bedeutsam sein. Beim aktuellen Systementwurf fließt in

jedem Fall das Praxiswissen des Entwerfers und von erfahrenen Benutzern im Umfeld der Systementwicklung ein.

Auch wenn diese Vorstellungen einer benutzerorientierten und aufgabenzentrierten Systementwicklung entsprechen, so ist doch ein weiterer wichtiger Faktor zu betonen: Anregungen und Randbedingungen aus den Computerwissenschaften und der allgemeinen technologischen Entwicklung. Man muß natürlich wissen, was technologisch möglich ist und wie die Vorstellungen umzusetzen sind. Diese Einflüsse kann man unter dem Stichwort *technologiebewußter* Systementwurf zusammenfassen. Aber – und da sind wir bei einem entscheidenden Punkt für Forschungsstrategien – wer sollte wem welche Vorgaben machen? An dieser Stelle erscheint es hilfreich, sich die z. Zt. existierenden Forschungsstrategien genauer anzuschauen. Dazu erweist sich eine Kategorisierung von Forschungs- und Entwicklungsstrategien im Hinblick auf methodische Aspekte als nützlich. So ist zwischen deskriptiven, prädiktiven, konstruktiven und evaluativen Vorgehensweisen zu unterscheiden. In unterschiedlicher Kombination finden diese Eingang in die folgenden drei Arten von Forschung /Streitz 86/:

### ○ Nachlaufforschung

Darunter fällt eigentlich der größte Teil der z. Zt. in der Software-Ergonomie durchgeführten psychologisch und arbeitswissenschaftlich orientierten Forschung. Bestehende Systeme werden *deskriptiv* analysiert und *nachträglich evaluiert*, vgl. /Oppermann, in diesem Band/, d. h. auf ihre Erlempbarkeit und Benutzbarkeit hin untersucht und auf ihre Übereinstimmung mit Normen – soweit vorhanden – überprüft. Dieses Vorgehen ist natürlich notwendig, da der Entwerfer dadurch Rückmeldung über die von ihm implementierten Systeme erhält. Außerdem ist es dabei auch möglich, existierende Systeme auf ihre *Verträglichkeit* mit Anforderungen z. B. aus der Wahrnehmungs- und Gedächtnispsychologie hin zu untersuchen und zu bewerten. Als entscheidender Nachteil ist festzuhalten, daß diese Forschung der technologischen Entwicklung immer hinterherlaufen muß.

### ○ Begleitforschung

Darunter verstehe ich die Teilnahme software-ergonomischer Experten im Entwurfsteam. Sie begleiten die aktuelle Entwicklung von Anfang an und beraten bei Entscheidungen über die Gestaltung der Mensch-Computer-Schnittstelle. Wir finden hier *deskriptive* und *evaluative* Verfahren auf der Seite des Ergonomen und des Psychologen gepaart mit einem *konstruktiven* Vorgehen auf der Seite des Entwerfers. Eine sinnvolle Hilfe ist dabei die Entwicklung von Prototypen, die bereits alle wesentlichen Charakteristika der Schnittstelle zeigen, aber noch keine umfangreiche Funktionalität aufweisen. Dadurch ist es zu einem frühen Zeitpunkt im Entwicklungszyklus möglich, Experimente durchzuführen und Änderungen mit geringem und damit vertretbarem Aufwand zu erreichen. Dieser Ansatz stellt bereits einen entscheidenden Fortschritt gegenüber der Nachlaufforschung dar und wird heutzutage an einigen Stellen auch praktiziert.

## ○ Vorlaufforschung

Die zuvor erwähnte Begleitforschung sollte m. E. durch den Ansatz der Vorlaufforschung ergänzt werden, der sehr stark auf psychologische Grundlagenforschung angewiesen ist. Die Grundidee besteht hier darin, daß aus den Erkenntnissen über die Art und Weise, wie Menschen ihre Arbeitsaufgaben und Problemstellungen bearbeiten und lösen, Vorgaben an zu entwickelnde Systeme und damit auch an Technologien abgeleitet werden. Diesen Vorgaben sollten Analysen zugrundeliegen, die sich zwar an den Eigenschaften menschlicher Informationsverarbeitung orientieren, sie aber nicht in ihren Beschränkungen simulieren. Damit ist die Vorlaufforschung prädiktiv und konstruktiv ausgerichtet. Sie dient dazu, die technische Entwicklung anzuregen, um Möglichkeiten bereitzustellen, die diese Beschränkungen zu überwinden helfen. Dabei sollte aber weiterhin das Prinzip der kognitiven Kompatibilität in bezug auf die jeweiligen Wissensrepräsentationen zwischen Benutzer und System gewährleistet sein.

## 1.7 Schlußbemerkung

Damit wir dem Ziel – der Entwicklung benutzerorientierter und aufgabenzentrierter Computersysteme – näherkommen, ist es notwendig, das erforderliche Wissen auf vielen Ebenen der Beteiligung an diesem Prozeß bereitzustellen. Da es aber bisher keinen Studiengang oder auch nur ein Curriculum *Software-Ergonomie* gibt, war es notwendig, dieses Wissen, das sehr verstreut und teilweise nur fachspezifisch formuliert vorliegt, in einer aufbereiteten Form zusammenzustellen. Dies geschieht nun in dem vorliegenden Buch, dessen Kapitel auf wiederholt überarbeiteten Beiträgen zur Software-Ergonomie Herbstschule (SEH) basieren. Die Software-Ergonomie Herbstschule wird seit 1986 jährlich als einwöchige Weiterbildungsveranstaltung im Rahmen der Deutschen Informatik Akademie (DIA) von der Fachgruppe *Software-Ergonomie* in der Gesellschaft für Informatik veranstaltet.

## Literatur

/Badre , Shneiderman 80/

Badre A., Shneiderman B. (Eds.): *Directions in human-computer interaction*, Norwood, N.J.: Ablex 1980

/Balzert 83/

Balzert H. (Hrsg.): *Software-Ergonomie*, Stuttgart: Teubner 1983

/Balzert 86/

Balzert H.: *Gestaltungsziele der Software-Ergonomie*, in: *Software-Ergonomie Herbstschule SEH '86*, Berlin 1986

/Balzert, in diesem Band/

Balzert H.: *E/A-Geräte für die Mensch-Computer-Interaktion*, in diesem Band

/Bullinger 85/

Bullinger H. (Hrsg.): *Software-Ergonomie '85 – Mensch-Computer Interaktion*. Stuttgart: Teubner 1985

- /Bullinger, Shackel 87/  
 Bullinger H., Shackel B. (Eds.): *Proceedings of the Second IFIP Conference – Human-computer interaction INTERACT '87*, Amsterdam: North-Holland 1987
- /Cakir, Hard, Stewart 80/  
 Cakir A., Hart D., Stewart, T.: *Visual display terminals. A manual*, New York: John Wiley & Sons 1980
- /Cakir, Hart, Stewart 80/  
 Cakir A., Hart D., Stewart T.: *Bildschirmarbeitsplätze*, Heidelberg: Springer 1980
- /Card, Moran, Newell 83/  
 Card S. K., Moran T. P., Newell A.: *The psychology of human-computer interaction*, Hillsdale, N. J.: Erlbaum 1983
- /Coombs, Alty 81/  
 Coombs M., Alty J. (Eds.): *Computing skills and the user interface*, London: Academic Press 1981
- /Dzida 80/  
 Dzida W.: *Kognitive Ergonomie für Bildschirmarbeitsplätze*, in: *Humane Produktion / Humane Arbeitsplätze*, 10, 1980, S.18-19
- /Falzon et al. 88/  
 Falzon P., Hoc J., Streitz N., Waern Y. (Eds.): *Cognitive Ergonomics – Understanding, learning, and designing human-computer interaction*. Chichester: John Wiley & Sons, in Vorbereitung.
- /Fährnich 87/  
 Fährnich K.: *Software-Ergonomie: Stand und Entwicklung*. in: K. Fährnich (Hrsg.): *Software-Ergonomie. State-of-the-Art 5*. München: Oldenbourg 1987, S. 9-28
- /Gaines 84/  
 Gaines B.: *From ergonomics to the fifth generation*, in: B. Shackel (Ed.): *Proceedings of the INTERACT'84 – Human-computer interaction*. Amsterdam: North-Holland 1984, pp. 3-7
- /Gaines 86/  
 Gaines B.: *From timesharing to the sixth generation: the development of human-computer interaction*, in: *International Journal of Man-Machine Studies*, 24, 1986, Part I, pp. 1-27
- /Gould, Lewis 85/  
 Gould J., Lewis C.: *Designing for usability: key principles and what designers think*, in: Shackel, B. (Ed.): *Proceedings of the Conference INTERACT '84 – Human-computer interaction*. Amsterdam: North-Holland 1985
- /Grandjean, Vigliani 80/  
 Grandjean E., Vigliani E. (Eds.): *Ergonomic aspects of visual display terminals*, London: Taylor & Francis 1980
- /Green, Payne, van der Veer 83/  
 Green T., Payne S., van der Veer G. (Eds.): *The psychology of computer use*. London: Academic Press 1983
- /Guliano 82/  
 Guliano V.E.: *The mechanization of office work*, in: *Scientific American*, 247, 1982, pp.148-165
- /Hacker 87/  
 Hacker W.: *Software-Gestaltung als Arbeitgestaltung*, in: K. Fährnich (Hrsg.): *Software-Ergonomie. State-of-the-Art 5*. München: Oldenbourg 1987, S. 29-42
- /Hacker, Schönfelder 86/  
 Hacker W., Schönfelder E.: *Job organization and allocation of functions between man and computer*, in: Klix F., Wandke H. (Eds.): *Man-Computer Interaction Research: MACINTER I*, Amsterdam: North-Holland 1986, pp. 403-420

/Hartson 85/

Hartson H.: *Advances in human-computer interaction*. Norwood, N.J.: Ablex 1985

/Klix et al. 88/

Klix F., Streitz N. A., Wandke H., Waern Y.: *Man-Computer Interaction Research, MACINTER II*, Amsterdam: North-Holland, in Vorbereitung

/Klix, Wandke 86/

Klix F., Wandke H. (Eds.): *Man-Computer Interaction Research, MACINTER I.*, Amsterdam: North-Holland 1986

/Klix, et al. 88/

Klix F., Streitz N., Waern Y., Wandke H. (Eds.): *Man - computer interaction research: MACINTER II.*, Amsterdam: North-Holland 1988

/Martin 73/

Martin J.: *Design of man-computer dialogues*, Engelwood Cliffs N.J.: Prentice Hall 1973

/Nickerson 69/

Nickerson R.: *Man-computer interaction: a challenge for human factors research*, in: IEEE Transactions on Man-Machine Systems, 4, 1969, pp. 164-180

/Norman 86/

Norman D.A.: *Cognitive Engineering*, in: Norman D. A., Draper S. (Eds.): *User centered system design: New perspectives on human-computer interaction*, Hillsdale N.J.: Erlbaum 1986, pp. 31-61

/Norman, Draper 86/

Norman D., Draper S. (Eds.): *User centered system design: New perspectives on human-computer interaction*, Hillsdale N. J.: Erlbaum 1986

/Oppermann, in diesem Band/

Oppermann R.: *Software-ergonomische Evaluationsforschung*, in diesem Band

/Rasmussen, 87/

Rasmussen J.: *Cognitive Engineering*, in: Bullinger H., Shackel B. (Ed.): *Proceedings of the Conference INTERACT'87 - Human-computer interaction*. Amsterdam: North-Holland 1987, pp. XXV-XXX

/Sackman 70/

Sackman H.: *Man-computer problem solving*, Princeton: Auerbach 1970

/Salvendy 84/

Salvendy G.: *Human-computer interaction*, Proceedings of First USA-Japan Conference. Amsterdam: Elsevier 1984

/Salvendy 87/

Salvendy G.: *Cognitive engineering in the design of human-computer interaction and expert systems*, Amsterdam: Elsevier 1987

/Salvendy, Sauter, Hurrell 87/

Salvendy, G. Sauter S., Hurrell J.: *Social, ergonomic, and stress aspects of work with computers*, Amsterdam: Elsevier 1987

/Schönpflug, Wittstock 87/

Schönpflug W., Wittstock M. (Hrsg.): *Software-Ergonomie '87 - Nützen Informationssysteme dem Benutzer?* Stuttgart: Teubner 1987

/Shackel 59/

Shackel B.: *Ergonomics for a computer*, in: Design, 120, 1959, pp.36-39

/Shackel 69/

Shackel B.: *Man-computer interaction: the contribution of the human sciences*, in: Ergonomics, 12, 1969, pp. 485-499

- /Shackel 81/  
Shackel B. (Ed.): *Man-computer interaction: Human factors aspects of computers and people*, The Netherlands: Sijthoff & Noordhoff 1981
- /Shackel 84/  
Shackel B. (Ed.): *Proceedings of the Conference INTERACT '84 – Human-computer interaction*, Amsterdam: North-Holland 1984
- /Shackel 85/  
Shackel B.: *Human factors and usability – whence and whither?* in: H. Bullinger (Hrsg.): *Software-Ergonomie '85 – Mensch-Computer-Interaktion*, Stuttgart: Teubner 1985, S.13-31
- /Shneiderman 80/  
Shneiderman B.: *Software psychology: Human factors in computer and information systems*, Boston: Little, Brown and Co. 1980
- /Shneiderman 87/  
Shneiderman B.: *Designing the user interface: Strategies for effective human-computer interaction*, Reading: Addison-Wesley 1987
- /Sime, Coombs 83/  
Sime M., Coombs M. (Eds.): *Designing for human-computer communication*, London: Academic Press 1983
- /Spinas, Troy, Ulich 83/  
Spinas Ph., Troy N., Ulich E.: *Leitfaden zur Einführung und Gestaltung von Arbeit an Bildschirmarbeitsplätzen*, Zürich: Verlag Industrielle Organisation 1983
- /Skarpelis 87/  
Skarpelis C.: *Software gestalten heißt Arbeitsbedingungen gestalten*, in: K. Fähnrich (Hrsg.): *Software-Ergonomie. State-of-the-Art 5*, München: Oldenbourg 1987, S. 54-71
- /Smith, Green 80/  
Smith H., Green T. (Eds.): *Human interaction with computers*, London: Academic Press 1980
- /Streitz 85/  
Streitz N. A.: *Die Rolle von mentalen und konzeptuellen Modellen in der Mensch-Computer Interaktion: Konsequenzen für die Software- Ergonomie?* in: H.J. Bullinger (Hrsg.): *Software-Ergonomie '85: Mensch-Computer-Interaktion*. Stuttgart: Teubner 1985, S. 280-292
- /Streitz 86/  
Streitz N. A.: *Cognitive Ergonomics: An approach for the design of user-oriented systems*, in: Klix F., Wandke H. (Eds.): *Man-Computer Interaction Research : MACINTER I*, Amsterdam: North-Holland 1986, pp.21-33
- /Streitz 87/  
Streitz N. A.: *Cognitive compatibility as a central issue in human-computer interaction: Theoretical framework and empirical findings.*, in: Salvendy, G. (Ed.): *Cognitive engineering in the design of human-computer interaction and expert systems*, Amsterdam: Elsevier 1987, pp. 75-82
- /Streitz 88a/  
Streitz N. A.: *Mental models and metaphors: Implications for the design of adaptive user-system interfaces*, in: Mandl H., Lesgold A. (Eds.): *Learning issues for intelligent tutoring systems*, Cognitive Science Series, New York: Springer 1988, pp. 164-186
- /Streitz 88b/  
Streitz N. A.: *Psychologische Aspekte der Mensch-Computer-Interaktion*, in: Hoyos, Zimolong (Hrsg.): *Ingenieurpsychologie. Enzyklopädie der Psychologie*. Göttingen: Hogrefe 1988
- /Streitz, Eberleh 88/  
Streitz N. A., Eberleh E. (Hrsg.): *Mentale Belastung und kognitive Prozesse bei komplexen Dialogstrukturen*, Abschlußbericht an die Bundesanstalt für Arbeitsschutz, Dortmund. (Arbeitsbericht Nr. 1/48 -1988 des Instituts für Psychologie der RWTH Aachen)

/Thomas 84/

Thomas J., Schneider M. (Eds.): *Human factors in computer systems*, Norwood, N.J.: Ablex 1984

/Ulich, in diesem Band/

Ulich, E.: *Arbeits- und organisationspsychologische Aspekte*, in diesem Band

/Vassiliou 84/

Vassiliou Y. (Ed.): *Human factors and interactive computer systems*, Norwood, N.J.: Ablex 1984

/van der Veer et al. 84/

van der Veer G., Tauber M., Green T., Gorny P. (Eds.): *Readings on cognitive ergonomics – mind and computers*, Berlin: Springer 1984

/Weinberg 71/

Weinberg G.: *The psychology of computer programming*, New York: Nostrand Reinhold 1971