

Transparenz - Eine zentrale Software-ergonomische Forderung

Susanne Maaß

Gliederung

0. Einleitung
1. Systemtransparenz in der Literatur
2. Menschengerechte Arbeit
3. Software-Ergonomie: Menschengerechte Arbeit mit Computern
4. Durchschauen, Beherrschen, Transparenz
5. Förderung von Transparenz
6. Zusammenfassung
7. Literatur

0. Einleitung

Computersysteme sollen transparent sein. Benutzer und Entwickler*) stimmen mit dieser Aussage überein. Spontan wird der Begriff der Transparenz gleichgesetzt mit Durchschaubarkeit, Verständlichkeit, Durchsichtigkeit. Ein transparentes System bleibt keine Black Box für den Benutzer; es öffnet sich und gewährt Einblick. Doch gibt es auch Widerspruch. Benutzer wollten gar nicht so viel wissen über das System, insbesondere nicht über seine interne Struktur, wird eingewendet. Die Interna sollten im Gegenteil gerade verdeckt werden, um die Benutzer nicht zu verwirren und zu überfordern. Eine offengelegte Systemstruktur verstelle den Benutzern den Blick auf ihr eigentliches Problem, bei dessen Lösung ihnen der Computer helfen sollte. Eigentlich müßte der Computer möglichst transparent im Sinne von durchsichtig und wenig störend sein, so daß man leicht hindurchsehen kann.

Ich vermute, daß diese widersprüchlichen Auffassungen aufgrund eines unterschiedlichen und ungeklärten Verständnisses des Begriffs Transparenz entstehen, und möchte im folgenden auf eine Definition hinarbeiten. Den Rahmen für diese Definition wird das sog. Kontrollkonzept liefern, das in der Arbeitswissenschaft als zentral für menschengerechte Arbeitsgestaltung angesehen wird. Der Arbeitende soll sich in seiner Umwelt orientieren und seine Arbeitsbedingungen und -prozesse selbst beeinflussen (kontrollieren) können. Transparenz wird von mir als Idealzustand der Mensch-Computer-Interaktion beschrieben, in dem eine Benutzerin das verwendete System individuell durchschaut, im Rahmen ihrer Arbeitsaufgaben zielgerichtet nutzen und an eigene Erfordernisse anpassen kann. Durch den Bezug auf das Kontrollkonzept wird deutlich, daß Transparenz als subjektives und kontextabhängiges Konzept verstanden werden muß.

Die Begriffe System, Computersystem und manchmal auch Computer werden im folgenden synonym verwendet. Im Kontext Software-ergonomischer Systemgestaltung wird unter "Computer-System" oder "System" meist ein spezielles Anwendungssystem oder integriertes System verstanden, das auf einem PC, einem Großrechner oder auf vernetzten

*) Es wird im folgenden nicht an allen Stellen von Benutzerinnen und Benutzern, Systementwicklerinnen und Systementwicklern gesprochen. Um trotzdem dem Eindruck entgegenzuwirken, daß Frauen im Kontext von Systementwicklung und -nutzung nicht vorkämen, werden stattdessen zeitweilig nur die weiblichen und zeitweilig nur die männlichen Bezeichner verwendet. Das jeweils andere Geschlecht ist immer mitgemeint.

Rechnern realisiert ist und dessen Benutzungsschnittstelle für den Gebrauch durch Einzelne angemessen zu gestalten ist. Erst neuerdings beginnt die Software-Ergonomie, sich mit Software zur Unterstützung von Gruppenarbeit zu beschäftigen und damit über die Betrachtung der Mensch-Computer-Dyade hinauszugehen. In diesem Fall ist das Verhalten des Computersystems gegenüber einem Benutzer nicht allein durch dessen eigene Eingaben und den Programmcode, sondern auch durch die Eingaben anderer Gruppenmitglieder bestimmt. Das System, mit dem der Benutzer konfrontiert ist, schließt also diese Menschen quasi mit ein. Die Bedeutung der auf Einzelarbeit bezogenen Software-ergonomischen Kriterien der Systemgestaltung müssen für diesen erweiterten Kontext ganz anders diskutiert werden, denn Gestaltung im Interesse jedes einzelnen kann den Interessen der jeweils anderen oder der ganzen Gruppe zuwiderlaufen. Dies gilt auch für die Forderung nach Transparenz. Ich werde mich im folgenden weitgehend auf die Betrachtung von Einzelarbeit am Rechner beschränken.

Die wenigen expliziten Definitionen von Transparenz, die in der Software-ergonomischen Literatur zu finden sind, werden in Abschnitt 1 zusammengestellt. Abschnitt 2 führt kurz das Kontrollkonzept als Rahmen für meine Betrachtungen von Transparenz ein. In Abschnitt 3 wird Kontrolle genauer im Kontext der Gestaltung computergestützter Arbeit diskutiert. Die enge Bindung des subjektiven Kontroll-Erlebens an die individuellen Zwecke der Benutzerinnen und Benutzer wird verdeutlicht.

Abschnitt 4 trifft zunächst eine Unterscheidung zwischen grundsätzlicher Durchschaubarkeit im Entwicklungskontext und individuellem Durchschauen beim Gebrauch. Darauf aufbauend werden die Begriffe Beherrschen und Transparenz definiert und diskutiert. Zwei in der Software-Ergonomie sehr populäre Konzepte, Direkte Manipulation und Konsistenz, werden dann in Abschnitt 5 exemplarisch auf ihre Transparenzförderlichkeit untersucht. Abschnitt 6 gibt eine abschließende Zusammenfassung.

1. Systemtransparenz in der Literatur

Die Forderung nach Systemtransparenz stammt aus dem Software-ergonomischen Umfeld. Die traditionelle Software-Ergonomie betrachtet die Arbeit Einzelner an Computersystemen. Ganz allgemein geht es darum, den Benutzerinnen und Benutzern ihre Arbeit mit dem Computersystem zu erleichtern. In der Literatur erfolgte bislang keine intensive Beschäftigung mit dem Begriff Transparenz. Diskussionen bewegen sich meist auf dem in der Einleitung dargestellten Niveau. Die wenigen genaueren Definitionen oder Charakterisierungen von Transparenz werden im folgenden zusammengestellt.

Wandmacher (1993, S. 204) nennt Transparenz als ein Bewertungsmerkmal von Benutzungsschnittstellen und setzt ihn mit Durchschaubarkeit gleich: "Durchschaubarkeit oder Transparenz bezeichnet den Aspekt der Selbstbeschreibungsfähigkeit, daß 'jeder einzelne Dialogschritt unmittelbar verständlich ist' (DIN 66234 Teil 8, S. 2)". Unter Bezug auf Fitter (1979) schreibt er: "Ein transparentes System funktioniert wie ein Automat mit leicht erkennbaren Zuständen, die der Benutzer in sein mentales Modell des Systems einordnen und als unmittelbares Ergebnis seiner Aktionen verstehen kann. (...) Zur Durchschaubarkeit gehört weiterhin, daß der Benutzer jederzeit erkennen kann, woher er kommt (vorheriger Zustand und vorherige Aktion), wo er sich gerade befindet (aktueller Zustand), und wohin er gehen kann. Die möglichen Aktionen, verfügbaren Funktionen und die vom aktuellen Zustand aus erreichbaren Zustände müssen erkennbar sein. Die Erkennbarkeit der Funktionalität der Benutzungsoberfläche ist damit ein Moment der Durchschaubarkeit." Für den Benutzer fordert er damit Antworten auf die sog. "Nievergelt'schen Fragen": Wo bin ich? Was kann ich hier tun? Wie kam ich hierhin? Wo kann ich hin und wie komme ich dorthin? (Nievergelt, 1983, S. 44) Der Benutzer soll also Möglichkeiten zur Orientierung in der dynamischen Interaktion mit dem System haben.

Spinas, Waeber und Strohm (1990, S.11 f.) sprechen explizit von der Unterstützung beim Aufbau eines mentalen Modells vom System: "Die Transparenz eines Systems soll dem Benutzer die Bildung eines Struktur- und Prozeßmodells des Systems im Gedächtnis er-

leichtern, was ihm die notwendigen Orientierungsgrundlagen für die Benutzung bietet." Er soll sich "über den Inhalt (Funktionen, Informationen) und dessen Organisation im System (Zugriffspfade, Ordnungskriterien) generell sowie über einen aktuellen Dialogzustand ein 'Bild' machen können, um die technische Unterstützung auch effizient nutzen zu können." Sowohl die Funktionalität, d.h. die Gesamtheit der Anwendungsfunktionen, die das System dem Benutzer bietet, als auch die Handhabung des Systems sollte der Benutzer also durchschauen können.

Beide Aspekte der Durchschaubarkeit wurden auch von mir in einem früheren Beitrag gefordert (Maaß, 1983, S. 25, frei ins Deutsche rückübersetzt): "Ein transparentes System erleichtert es dem Benutzer, sich ein inneres Modell von den Funktionen zu machen, die das System für ihn ausführen kann. Dies schließt sowohl die Anwendungsfunktionen als auch die Handhabungsfunktionen ein." Und weiter: "Die Benutzungsschnittstelle ist auf das Vorwissen des Benutzers über den Anwendungsbereich und seine menschliche Intuition (insb. seine Kommunikationsgewohnheiten) abgestimmt. Der Aufwand für die Systemhandhabung darf die Lösung des eigentlichen Problems im Anwendungsbereich nicht stören." Im zweiten Teil dieser Definition wird eine Beziehung hergestellt zwischen der Transparenz eines Systems und dem Vorwissen bzw. den Gewohnheiten der Benutzerinnen und Benutzer: Je angepaßter ein System in dieser Hinsicht ist, desto selbstverständlicher ist es zu benutzen.

Auf eben diesen Effekt beziehen sich auch Holtzblatt, Jones und Good (1988, S. 46). Sie berichten über ihre Beobachtungen von Benutzerinnen und Benutzern: "We see the experience of transparency when users remain in the flow of their work and are not disrupted by the computer system. Users of more transparent systems focus on the accomplishment of their tasks, and feel satisfied with how their work is moving along. The focus of users' attention is not diverted to the use of the interface." In dieser Variante erhält Systemtransparenz die Bedeutung von "Durchsichtigkeit".

Holtzblatt et al. weisen auf einige verwandte Auffassungen hin. So schreibt Rutkowski (1982, S. 299 f.) unter der Überschrift "The Computer is a tool": "In an ideal situation the relationship of user and tool approaches one of transparency. The user is able to apply intellect directly to the task; the tool itself seems to disappear. This transparency is characteristic of all expert applications of tools - from hacksaws to racing cars." Auch Laurel (1986, S. 76 f.) fordert für den Benutzer das Gefühl, selbst direkt im vom Computer dargestellten Anwendungskontext zu handeln, ohne Zwischenschaltung eines Werkzeuges oder Vermittlers, und nennt dies "first-personness". Hutchins, Hollan und Norman (1986) prägen die Bezeichnung "direct engagement" für dieses Phänomen.

Auch Bødker sieht es als Idealfall an, wenn die Benutzerinnen vergessen, daß zwischen ihnen und ihren Arbeitsgegenständen der Computer steht. Sie arbeiten dann nicht *mit* der, sondern *durch* die Benutzungsschnittstelle (hindurch) auf den Gegenständen: "Through the interface" (Bødker, 1991).

Winograd und Flores (1986) unterscheiden zwei Arten des Umgangs mit Dingen: Im normalen Gebrauch sind sie zuhanden ("ready-to-hand") und erfordern keine bewußte Aufmerksamkeit. Erst im Falle einer Störung ("break down"), einem Zustand der Nichtoffen-sichtlichkeit, rücken sie ins Bewußtsein, sind als eigenständige Objekte mit besonderen Eigenschaften vorhanden ("present-at-hand") und werden reflektiert. Zu dieser Begrifflichkeit ließen sie sich anregen durch gleichlautende Begriffe bei Heidegger (1984), die sie im neuen Kontext - Gestaltung von Computer-Systemen - erweitert interpretieren. Wenn Winograd und Flores eine "Transparenz der Interaktion" anstreben (S. 164), so meinen sie damit, daß die Verständigung in der Interaktion zwischen Mensch und Computer mit minimalem bewußtem Aufwand erzielt wird. Für das Design von Werkzeugen wird die Zuhandenheit gefordert. Allerdings sind Störungen nicht von vornherein auszuschließen. Gutes Design antizipiert Formen von Störungen und gestaltet für den Benutzer einen angemessenen Raum für mögliche Handlungen in solchen Situationen, es ermöglicht Orientierung und Lernen.

Herrmann und Wulf (beide 1994) beschäftigen sich ausführlich mit der Frage der System-

transparenz. Allerdings führen sie ihre Diskussion mit besonderem Bezug auf Groupware-Systeme und ISDN-Telefonie zur Unterstützung von Kommunikation und Kooperation in Gruppen und gehen damit über den traditionellen Blickwinkel der Software-Ergonomie hinaus. Für jedes Gruppenmitglied ist es wichtig zu wissen, wie die anderen das System nutzen und welche Einstellungen sie am System vorgenommen haben. Von Herrmann und Wulf wird zwischen funktionaler und nutzungsbezogener Transparenz unterschieden. Funktionale Transparenz bedeutet, daß "die Wirkung von Funktionen für ihre Nutzer nachvollziehbar wird. ... Die Nutzer sollen die prinzipiell zur Verfügung stehenden Funktionsalternativen erkennen können." (Wulf, 1994, S.129) "Nutzungsbezogene Transparenz macht den (aktuellen, S.M.) Gebrauch einzelner Grundfunktionen ... technisch nachvollziehbar" (a.a.O., S.131), d.h. es wird offengelegt, welcher Benutzer welche Funktionen wie gebraucht. Es ist offensichtlich, daß die Transparenzforderung in diesem Kontext konfliktträchtig wird, da sich die Interessen der betroffenen Gruppenmitglieder widersprechen können. Eine befriedigende Lösung kann nur durch Aushandlungsprozesse erreicht werden.

Ich fasse zusammen: Im Kontext der traditionellen Software-Ergonomie wird Transparenz meist als Systemeigenschaft verstanden, die den Aufbau eines mentalen Modells und damit die Orientierung im System erleichtert. Vereinzelt wird Transparenz auch als Eigenschaft der Interaktion oder der Beziehung zwischen Mensch und Computer beschrieben. Dabei wird abgehoben auf die ideale Übereinstimmung zwischen mentalem Modell und Systemverhalten; das System soll gewissermaßen durchsichtig für den Benutzer werden und nicht mehr störend in sein Bewußtsein dringen, so daß die Aufmerksamkeit primär auf die Erledigung der Aufgabe gerichtet werden kann.

In einigen dieser Definitionen steht Transparenz für die Forderung nach gutem Design schlechthin, in anderen als *ein* wichtiger Aspekt von Benutzbarkeit. Sicher ist, daß ein enger Bezug zur menschengerechten Arbeitsgestaltung besteht. Diesen Bezug möchte ich im folgenden genauer herausarbeiten und damit das Band weben, das die verschiedenen zitierten Auffassungen von Transparenz zusammenhält. Die Frage der Transparenz von Groupware soll ausgeklammert bleiben.

2. Menschengerechte Arbeit

Die Arbeitspsychologie hat für die Gestaltung menschengerechter Arbeit vier allgemeine Kriterien formuliert: Ausführbarkeit, Schädigungsfreiheit, Beeinträchtigungslosigkeit und Persönlichkeitsförderlichkeit. Die Arbeit muß grundsätzlich für den Menschen ausführbar sein. Das zweite und dritte Kriterium formuliert einen Schutzanspruch: die Arbeit soll den arbeitenden Menschen körperlich und psychisch weder dauerhaft schädigen noch vorübergehend beeinträchtigen. Darüberhinaus wird gefordert, daß die Arbeit den Menschen möglichst in seiner persönlichen Entwicklung fördern soll.

Neuere arbeitspsychologische Theorien orientieren sich an einem Menschenbild, das den Menschen als ein aktiv auf seine Umwelt einwirkendes Wesen sieht, das sich selbst Ziele setzt, Hypothesen (Erwartungen) über seine Umwelt aufstellt, diese überprüft und entsprechend seinen Zielen und Hypothesen planmäßig handelt (Troy, 1981, S. 14). Zwei Grundmotive werden im Menschen angenommen: das Bedürfnis, ein zusammenhängendes sinnvolles Verständnis unserer Welt zu erhalten, sowie das Bedürfnis nach Kontrolle der Umwelt (a.a.O., S. 48). Man geht davon aus, daß der Mensch sich mentale Modelle von seiner Umwelt und seinen Handlungsmöglichkeiten darin bildet (vgl. Dutke, 1994). Angemessene mentale Modelle sind notwendig, um Handlungen zu planen und zu steuern. Auf ihnen basierend können Umweltbedingungen gezielt verändert werden, sie können "kontrolliert" werden.

Kontrolle ist ein zentrales Konzept für die menschengerechte Gestaltung von Arbeit. Semmer (1990, S. 190) schreibt: "Handlungs- bzw. Tätigkeitsspielraum, Freiheitsgrade, Kontrolle, Autonomie, Job decision latitude - so vielfältig die Terminologie und so nuancen-

reich die Konzepte auch sind: In der einschlägigen Literatur herrscht große Einmütigkeit, daß die Möglichkeit, Einfluß auf seine Angelegenheiten zu nehmen, über möglichst viele Aspekte seines Lebens - und somit auch seiner Arbeit - selbst zu entscheiden oder zumindest mitzuentcheiden, zu den Kriterien einer menschenwürdigen Lebensführung im allgemeinen wie einer persönlichkeitsförderlichen Arbeitsgestaltung im besonderen zu zählen ist." (Zitiert nach Ulich, 1991, S. 142.)

Nach Troy werden drei Stufen der Kontrolle unterschieden (Troy, 1981, S. 48 ff.). Die niedrigste Stufe ist die *Durchschaubarkeit* von Sachverhalten und Zusammenhängen. Durchschaubarkeit wird gleichgesetzt mit dem "Verständnis von für das Individuum bedeutsamen Zusammenhängen" und der nachträglichen "Erklärung von Ereignissen, Handlungsergebnissen usw." Sind die Dinge durchschaubar, so kann ein angemessenes mentales Modell davon entwickelt werden. Durchschaubarkeit ist die Voraussetzung für die nächste Stufe der Kontrolle, die *Vorhersehbarkeit*: Auf dieser Stufe werden Hypothesen und Prognosen über zukünftige Prozesse und Wirkungen des eigenen Handelns gebildet. Diese beiden ersten Stufen werden meist unter dem Begriff *Orientierung* zusammengefaßt. Ich werde im folgenden immer nur mit diesem Begriff operieren und nicht weiter zwischen Durchschaubarkeit und Vorhersehbarkeit unterscheiden.

Orientierung ist die Voraussetzung für die höchste Stufe der Kontrolle, die *Beeinflußbarkeit* von Bedingungen und Prozessen der Umwelt gemäß eigenen Zielen. Erst mit der Beeinflußbarkeit ist vollständige Kontrolle gegeben. Orientierung und Beeinflußbarkeit sind als zwei Stufen der Kontrolle relativ genau unterscheidbar. Insofern kann (in Abb. 1) tatsächlich von einem zunehmenden Grad der Kontrolle gesprochen werden. Wie weitgehend etwas durchschaut und vorhergesehen wird, und auch der Umfang, in dem Beeinflussung möglich ist, kann darüberhinaus jeweils variieren.

Im Zusammenhang der Streßforschung wird die individuelle Einschätzung der Kontrollierbarkeit von Umweltbedingungen für besonders wichtig gehalten: Je höher der Grad der Kontrolle eingeschätzt wird, desto weniger Streß können die Bedingungen auslösen. Wiederholter oder längerfristiger Kontrollverlust führt zu Streß und kann sogar psychische Krankheit nach sich ziehen. Die individuelle Erfahrung von Kontrolle oder Nicht-Kontrolle prägt darüberhinaus die zukünftigen Kontrollerwartungen einer Person und fördert bzw. lähmt ihre zielgerichtete planvolle Aktivität (vgl. Frese, 1987, S. 325 f.).

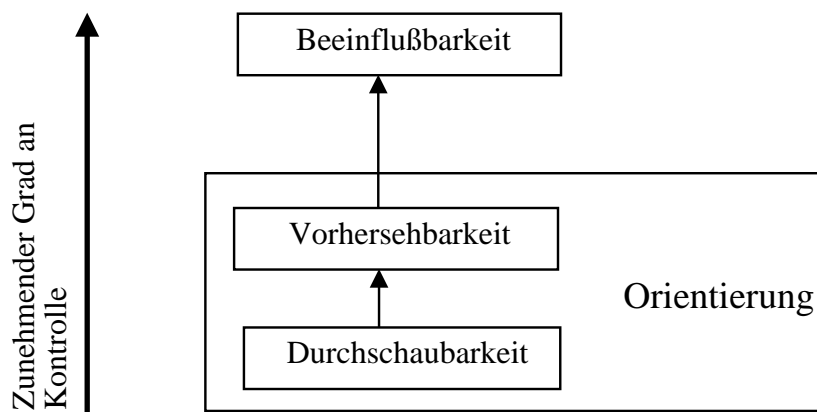


Abb. 1: Stufen der Kontrolle (in Anlehnung an Troy, 1981, S. 49)

Ich fasse zusammen: Im Hinblick auf menschengerechte Arbeitsgestaltung wird ein hohes Maß an Kontrolle für den Arbeitenden als wünschenswert angesehen. Er soll sich in seiner Umwelt orientieren und Bedingungen und Prozesse darin beeinflussen können. Handlungsspielräume haben einen nachhaltigen Einfluß auf seine Arbeitszufriedenheit.

Bei computergestützter Arbeit können Handlungsspielräume einerseits vernichtet, andererseits auch neu geschaffen werden. Die Nutzung von Spielräumen setzt voraus, daß sie zuvor erkannt werden. Die hohe Komplexität von technischen Systemen kann das Erkennen der Spielräume und damit die Kontrolle erschweren. Für den Benutzer undurchsichtige Komplexität ist eine häufige Ursache für mangelnden Benutzbarkeit von Computersystemen und ist damit zentrales Thema der Software-Ergonomie.

3. Software-Ergonomie: Menschengerechte Arbeit mit Computern

Die Software-Ergonomie beschäftigt sich mit der menschengerechten Gestaltung computergestützter Arbeit und insbesondere mit der angemessenen Gestaltung der Mensch-Computer-Schnittstelle. Sie sieht vier Faktoren als bestimmend für die Arbeitssituation von Benutzern und Benutzerinnen an: das zur Verfügung stehende technische System, die gegebene Aufgabe, die organisatorische Einbindung der Arbeit und die individuellen Gegebenheiten bei der Benutzerin selbst. Diese Faktoren stehen in enger Abhängigkeit zueinander. Menschengerechte Gestaltung computergestützter Arbeit muß Maßnahmen in allen vier Bereichen aufeinander abstimmen. Das Geflecht der vier Faktoren wird oft als modifizierte Leavitt-Raute (Oberquelle 1991) dargestellt oder vereinfacht im sog. ABC-Modell (vgl. auch Frese & Brodbeck, 1989) erfaßt, das den organisatorischen Kontext nicht extra ausweist (siehe Abb. 2).

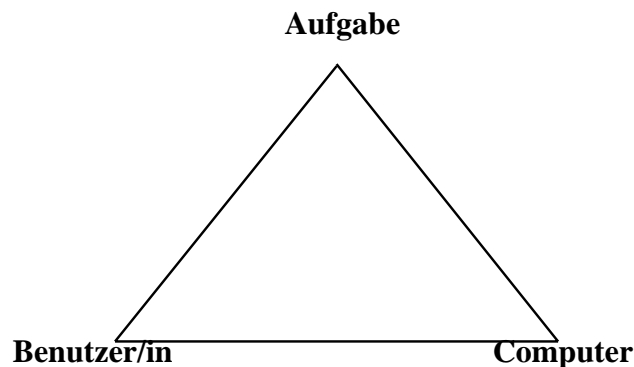


Abb. 2: Faktoren Software-ergonomischer Gestaltung im ABC-Modell *)

In der Vergangenheit sind eine Reihe von Software-ergonomischen Kriterien für die Analyse, Bewertung und Konstruktion von Systemen entwickelt worden. Dies sind z.B. die "Grundsätze ergonomischer Dialoggestaltung" und die "Dialogue Principles" (DIN 66234 Teil 8 und ISO 9241 Part 10). Diese Kriterien weisen einen engen Zusammenhang zu den in Abschnitt 2 genannten arbeitspsychologischen Kriterien menschengerechter Arbeit auf (vgl. Abb. 3). Koch, Reiterer und Tjoa (1991, S. 63) verstehen sie insbesondere als Ausdifferenzierung der Forderung nach Lern- und Entwicklungsmöglichkeiten, also nach Persönlichkeitsförderlichkeit für den arbeitenden Menschen. Die Aufgabenangemessenheit von Systemen und ihre Erlernbarkeit sehen sie dabei als übergeordnete Kategorien, die jeweils durch die übrigen Software-ergonomischen Kriterien realisiert werden.

Ein besonderes Problem entsteht bei der computergestützten Arbeit durch die für die Be-

*) Ursprünglich entstand die Dreiecksdarstellung in Diskussionen einer Arbeitsgruppe zum Thema "Transparenz von Dialogsystemen" bei der Tagung Mensch-Maschine-Kommunikation 1987. Der darüber erstellte Abschlußbericht (Keil-Slawik & Holl, 1987) enthält die Abbildung zum ersten Mal.

nutzerin oft undurchschaubare Komplexität des benutzten Computersystems. Nach Koch et al. erlauben einige der Software-ergonomischen Kriterien eine Orientierung der Benutzerin (Durchschauen, Vorhersehen), während andere eine Beeinflussung möglich machen. Die Orientierung ist Voraussetzung für die Beeinflussung. Abb. 3 zeigt, wie sich die Software-ergonomischen Kriterien in das Kontrollkonzept einbetten lassen.

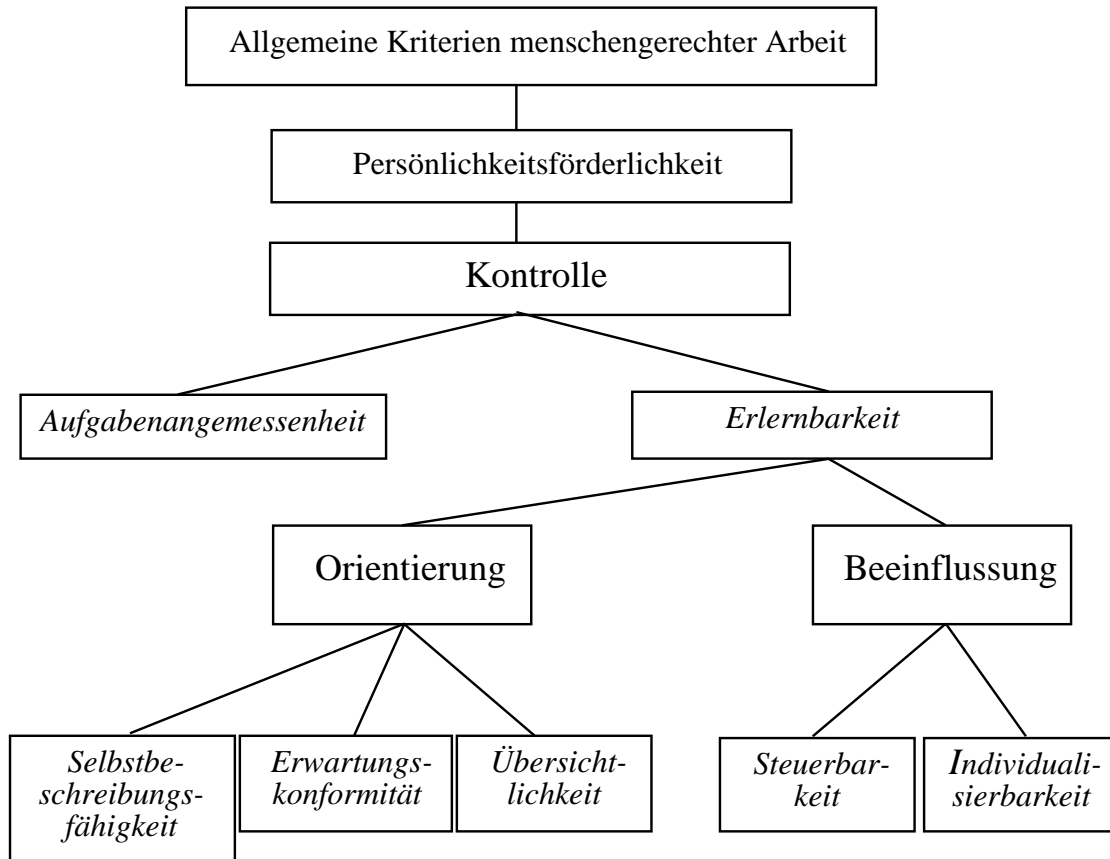


Abb. 3: Einbettung der *Software-ergonomischen Kriterien* in das Kontrollkonzept (in Anlehnung an eine Abbildung in Koch, Reiterer & Tjoa, 1991, S. 85)

Das Kontrollkonzept wurde für den Kontext computergestützter Arbeit von Frese (1987, S. 315 ff.) genauer beschrieben. Frese beleuchtet insbesondere die Beziehung zwischen Kontrolle und Komplexität und bezieht diese Konzepte auf die individuellen Ziele der Arbeitenden. Für menschengerechte Arbeit, d.h. im Sinne einer Humanisierung und Demokratisierung der Arbeit werden in der Arbeitswissenschaft generell größere Vielfalt und Spielräume für die Arbeitenden für wichtig gehalten. Dies erfordert in der Regel die Gestaltung von technischen Unterstützungssystemen mit vielfältiger und umfangreicher Funktionalität, in diesem Fall also komplexere Computersysteme. Diese Komplexität entsteht zunächst für die Entwicklerinnen auf der Ebene der programmtechnischen Realisierung. Für Benutzer muß die vielfältige Funktionalität dann zugreifbar gemacht werden. Dazu sind die für den Benutzer relevanten Teile der komplexen Struktur mittels der Benutzungsschnittstelle nach außen sichtbar zu machen. Biedenkopf (1994) bezeichnet dieses als das notwendige "Management der Komplexität".

Komplexität ist offensichtlich kein objektives Systemmerkmal. Abhängig von der Perspektive, die ein Betrachter einnimmt, kann die Einschätzung der Komplexität variieren. Im

Entwicklungskontext gesehen kann ein System überaus komplex und im Gebrauch gleichzeitig recht einfach sein; umgekehrt können allerdings auch wenig komplexe Systeme relativ kompliziert zu handhaben sein. Anzustreben ist immer eine einfache Gestaltung des Umgangs mit komplexen Systemen. Nach Biedenkopf führt höhere innere Komplexität allerdings meist zu komplizierter Außendarstellung und damit auch zu kompliziertem Zugriff.

Auch nach Frese (1987, S. 320 ff.) ist Komplexität keine reine Systemeigenschaft. Komplexität ist eine Eigenschaft der Interaktion zwischen einer Person und ihrer Umgebung. Im Kontext der Arbeit mit Computersystemen sieht er Komplexität in Abhängigkeit von den Zielen, die vom Benutzer zu verfolgen, den Plänen, die zu machen, und den Zeichen und Reaktionen des Systems, die zu interpretieren sind. Durch die Anwendungs- und Bedienfunktionen eines Systems sind die zulässigen bzw. notwendigen Ziele, Pläne und Interpretationen von Zeichen festgelegt. Mit deren Anzahl und mit der Menge und Art ihrer Beziehungen untereinander steigt die Komplexität eines Systems für die Benutzer. Frese spricht von systembedingt entstehenden Entscheidungsnotwendigkeiten im Handlungsablauf. Anders gesagt steigt die Komplexität des Umgangs mit der Anzahl von Entscheidungen, die getroffen werden müssen, und dem Grad ihrer Verwobenheit untereinander.

Die Bewältigung von Systemkomplexität durch die Benutzerin wird auch als Kontrolle bezeichnet. Bei der Arbeit mit einem Computersystem muß eine Benutzerin das System zumindest partiell durchschauen, um es für ihre eigenen Zwecke nutzen zu können. Frese differenziert zwischen internen und externen Voraussetzungen für Kontrolle (vgl. auch Abb. 4).

Als interne Voraussetzungen muß die Benutzerin hinreichende Kenntnisse und Fähigkeiten erwerben: Sie muß ein zweckmäßiges mentales Modell vom System entwickeln und die systembedingten Entscheidungsnotwendigkeiten kennen. Welche Ziele können überhaupt verfolgt werden? Was ist zu welchem Zeitpunkt zu tun? Welche Reihenfolgen sind einzuhalten? Wie werden Eingaben interpretiert, wie sind Systemausgaben zu interpretieren? Je komplexer der Umgang mit einem System ist, desto schwieriger ist es für die Benutzerin, diese "Orientierung" (erste Stufe der Kontrolle) zu bekommen.

Externe Voraussetzungen für Kontrolle sind nach Frese gegeben, wenn es innerhalb der und zusätzlich zu den Entscheidungsnotwendigkeiten auch Entscheidungsspielräume für die Benutzerin gibt. Wenn sie die Möglichkeit hat, sich neue und andere Ziele zu setzen, Pläne zu machen und Zeichen zu definieren und zu interpretieren, wenn sie das System also ihren eigenen Zielen anpassen kann, so verfügt sie über die zweite Stufe von Kontrolle: nicht nur Orientierung, sondern Einfluß.

Der Einfluß ist nicht allein darauf beschränkt, daß ein Benutzer, der sich mit dem System auskennt, Eingaben machen kann und somit die Abläufe im System steuert. Einfluß meint die Möglichkeit zur Veränderung der Systemfunktionalität und -handhabung durch die Benutzer. Natürlich soll es sich dabei nicht um eine Veränderung des Systemcodes handeln (Programmierung), sondern um eine Ausnutzung von vorgegebenen Varianten oder Modifikationsoptionen. Bietet eine Arbeitsumgebung keine oder nur geringe Spielräume dieser Art, so haben die Arbeitenden nicht die optimale Kontrolle. An dieser Formulierung erkennt man, daß die Kontrollfrage nicht einfach mit ja oder nein zu beantworten ist: Kontrolle ist graduell abgestuft vorhanden, denn der Einfluß der Benutzer kann mehr oder weniger umfangreich sein.

Die Möglichkeiten eines Benutzers, sich zu orientieren und Einfluß auszuüben, werden nach Frese maßgeblich durch die Funktionalität des Systems, seine Durchschaubarkeit und die Vorhersehbarkeit seines Verhaltens bestimmt. Diese Begriffe definiert er nicht genauer. Ich werde sie in Abb. 4 unter der Bezeichnung "Computersystem" zusammenfassen. Die Systementwickler stellen den Benutzern eine vielfältige Funktionalität und Spielräume zur Anpassung des Systems an eigene Erfordernisse zur Verfügung. Damit sind Absichten und Erwartungen verbunden, wie das System genutzt werden soll. So ergibt sich ein von

den Entwicklern angestrebter Grad an Kontrolle für die Benutzer.

Jedoch setzt sich nicht dieser Grad an Kontrolle, sondern das *subjektiv erlebte Maß an Kontrolle* in Leistungen und begleitende Streßeffekte um. Dieses subjektive Erleben geschieht vor dem Hintergrund individueller Ziele und Wertungen sowie der individuellen Kontrollerfahrungen in der Vergangenheit.

- Kontrolle ist immer vor dem Hintergrund der individuell verfolgten Ziele zu sehen. Nur Entscheidungsmöglichkeiten bzgl. Zielen, die ein Handelnder tatsächlich verfolgt, wirken sich auf sein Kontroll-Erleben aus.
- (Fehlende) Kontrolle wirkt sich unterschiedlich aus, je nachdem wo die betreffenden Ziele in der persönlichen Zielehierarchie der handelnden Person angesiedelt sind. Wichtige Ziele wiegen schwerer.
- Entscheidungsspielräume sind nur dann als positiv zu bewerten, wenn die einzelnen Entscheidungen keine hohen Risiken bergen. Riskante Entscheidungen werden vermieden und stellen daher keine Erweiterung der Handlungsmöglichkeiten dar.
- Je länger sich eine Person in Situationen der Kontrolle oder Nicht-Kontrolle befindet, desto stärker ist der Einfluß auf das erlebte Maß an Kontrolle. Kontrollerfahrungen führen zu entsprechenden Kontrollerwartungen für die Zukunft.

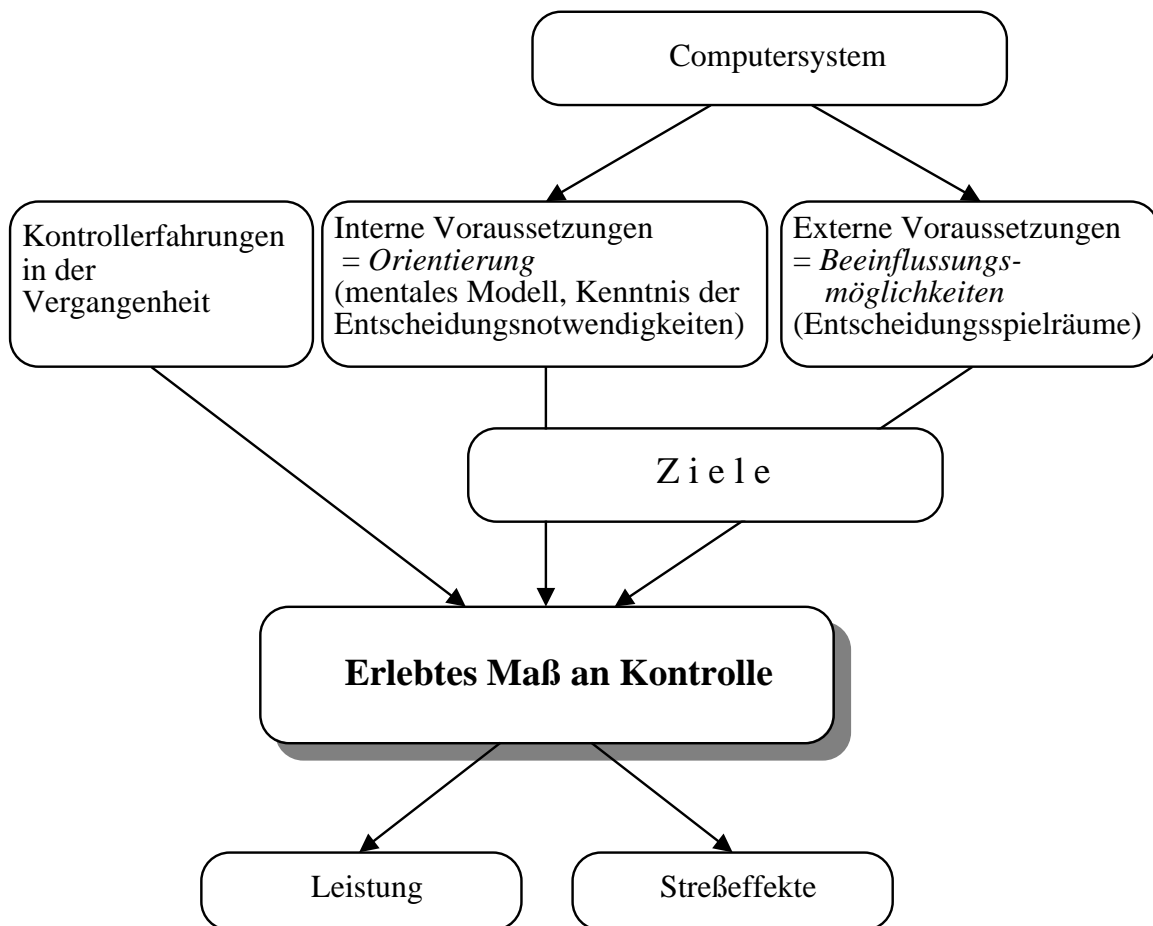


Abb. 4: Faktoren des subjektiven Kontrollerlebens bei computergestützter Arbeit in Abwandlung von Frese (1987, S. 315)

Ich komme zurück auf das Spannungsverhältnis zwischen Vielfalt und damit einhergehender Komplexität auf der einen Seite und Kontrolle auf der anderen: Wer Handlungsspielräume für den Benutzer fordert, handelt sich Komplexität ein. Komplexe Systeme sind jedoch schwerer zu durchschauen und zu beeinflussen für den Benutzer. Gleichzeitig wird aber möglichst weitgehende Kontrolle für den Benutzer gefordert. Wie geht man mit diesem Zielkonflikt um? Frese löst ihn auf, indem er den unerwünschten Teil der Komplexität gesondert als *Kompliziertheit* bezeichnet. Kompliziertheit ist, kurz gesagt, der Teil der Komplexität, der schwer zu kontrollieren ist. Kompliziertheit entsteht z.B., wenn die Funktionalität nicht aufgabenangemessen ist, wenn das System undurchschaubar ist und seine Reaktionen unvorhersehbar sind. Auch wenn es zu viele Entscheidungsnotwendigkeiten oder zu geringe Entscheidungsspielräume gibt, wird ein System als kompliziert empfunden.

Große Teile der Software-Ergonomie zielen darauf ab, Kompliziertheit für den Benutzer zu reduzieren. Komplexität von Arbeitssituationen ganz allgemein ist gar nicht zu vermeiden bzw. im Sinne qualifizierter und persönlichkeitsförderlicher Arbeit sogar erwünscht. Deswegen stellt auch die Software-Ergonomie Forderungen, die den Benutzern und Benutzerinnen Spielräume eröffnen sollen, gleichzeitig aber auf eine Erhöhung der Systemkomplexität hinauslaufen: z.B. vielfältige, aufgabenangemessene Funktionalität, Steuerbarkeit und Individualisierbarkeit.

Ich fasse die Ausführungen der Abschnitte 2 und 3 zusammen: Als zentrales Konzept für die Herstellung persönlichkeitsförderlicher Arbeitsbedingungen wurde das Kontrollkonzept dargestellt. Bei computergestützter Arbeit wird Kontrolle, d.h. Orientierung und Beeinflussung, durch die Komplexität der eingesetzten Programme erschwert. Allerdings ist diese Komplexität gleichzeitig ein Ausdruck der Handlungsspielräume, die - neben den Möglichkeiten der Kontrolle - für menschengerechte Arbeit gefordert werden. Im Sinne menschengerechter Arbeitsgestaltung muß es also darum gehen, unvermeidliche Komplexität durchschaubar zu machen, so daß die damit verbundenen Spielräume für die Benutzer und Benutzerinnen nutzbar werden. Unnötige Komplexität ist zu vermeiden. Allerdings wirken sich immer nur diejenigen Systemaspekte aus, die für die Zielsetzungen eines Benutzers oder einer Benutzerin relevant sind, und müssen für sie durchschaubar und beeinflussbar sein.

Auf dieser Grundlage möchte ich auf die Frage einer Definition von Transparenz zurückkommen.

4. Durchschauen, Beherrschen, Transparenz

Systeme können auf verschiedenen Abstraktionsebenen und aus verschiedenen Perspektiven durchschaut werden. Ich möchte hier zunächst die Perspektiven der Entwicklerinnen und Benutzerinnen unterscheiden.

Als von Menschen konstruiertes Artefakt ist jedes Computersystem grundsätzlich aus Entwicklerinnensicht durchschaubar. Dieses Durchschauen kann sich sowohl auf die physikalische Ebene (elektronische Bauelemente und Schaltvorgänge) als auch auf abstraktere Ebenen (z.B. Programmmoduln, Kontroll- und Datenfluß) beziehen. Sowohl der logische Aufbau als auch die dynamische Ablaufstruktur sind durch Programme und andere formale Beschreibungen wohl definiert. Jeder Systembestandteil, jeder Prozeß hat aus der Sicht der Entwicklerinnen und innerhalb der Systemlogik einen bestimmten Zweck. Der konkrete Bezug zu irgendwelchen Arbeitsabläufen oder Arbeitsgegenständen in einem Anwendungsbereich spielt aber beim Durchschauen nur eine untergeordnete Rolle. Im Entwicklungskontext wird vorwiegend in technischen und formalen Konzepten gedacht.

Grundsätzliche Durchschaubarkeit ist ein eher theoretisches Konzept und keine gestalterische Forderung im Interesse der Benutzer. Es ist zwar die politische Forderung zu stellen, daß Systeme nur dann eingesetzt werden sollten, wenn es Personen gibt, die aufgrund ihres umfassenden Systemverständnisses auch die Verantwortung für seinen Einsatz übernehmen

men können. Tatsächlich kennen aber meist sogar die Entwickler jeweils nur den Teil genauer, für den sie selbst zuständig waren, und kennen grob seine Einbettung in das Gesamtsystem. Die vielen Arten des Zusammenwirkens der Teile, wie sie sich im Gebrauch ergeben, sind theoretisch durchschaubar. Praktisch kann dieses Durchschauen nur noch kollektiv, also in der Gruppe aller beteiligten Entwickler geschehen und ist kaum durchführbar. Durch die technische Vernetzung und Verteilung von Systemen wird die Problematik noch weiter verschärft.

Wie ist auf diesem Hintergrund die Forderung nach Kontrolle und damit nach Durchschaubarkeit für die Benutzer zu verstehen? Benutzer brauchen die Systeme, mit denen sie arbeiten, nicht auf der gleichen Ebene wie die Entwickler zu durchschauen, denn für sie ist der Gebrauchszusammenhang ein ganz anderer. Für die Entwickler ist das System Arbeitsgegenstand und -produkt, das sie zu jedem Zeitpunkt im Erstellungsprozeß durchschauen müssen, um es modifizieren und weiterentwickeln zu können. Für Benutzer ist es ein komplexes Arbeitsmittel, das sie für ihre Anwendungszwecke einsetzen und in diesem Rahmen partiell durchschauen müssen.

Deshalb wird hier in eingeschränkter Weise eine *individuelle Durchschaubarkeit beim Gebrauch* gefordert. Der Computer muß vom individuellen Benutzer im Rahmen seiner Arbeit durchschaut werden.

Definition: "Individuelles Durchschauen beim Gebrauch"

Ein Computersystem wird von einer Benutzerin *individuell beim Gebrauch durchschaut*, wenn die Benutzerin die für ihren Arbeitskontext relevanten Konzepte von Funktionalität und Handhabung kennt, d.h. wenn sie sich ein zweckmäßiges mentales Modell vom System gemacht hat.

Zweckmäßig ist ein mentales Modell, wenn es für die aufgabenspezifischen Zwecke der Benutzerin ausreicht, so daß sie ihre Handlungen planen und die Systemausgaben verstehen kann.

Im Gegensatz zum Durchschauen im Entwicklungskontext geschieht dieses individuelle Durchschauen aus einer Anwendungsperspektive. Der Benutzer stellt den Zusammenhang her zwischen seiner Arbeitsaufgabe im Anwendungsbereich und den Systemobjekten und -funktionen: Wie dient das System meinen Zwecken, wie hilft es mir, meine Aufgaben zu bearbeiten, welche Arbeiten nimmt es mir ab? Wie stoße ich die entsprechenden Prozesse an?

Individuelles Durchschauen entspricht der ersten Stufe des Kontrollkonzeptes: Orientierung. Ein System ist genau dann individuell durchschaubar für einen Benutzer, wenn es ihm die für seinen Arbeitskontext nötige Orientierung ermöglicht. Mit dieser Orientierung sind die internen Voraussetzungen für weitergehende Kontrolle gegeben.

Die Definition läßt offen, ob die Benutzer die Zwecke erkennen können, die die Entwickler mit den einzelnen Systemfunktionen und mit dem System als ganzem verfolgt hatten. Wichtig ist nur der Bezug zu ihren eigenen Zielen. Da Kontrolle immer relativ zu angestrebten Zielen erlebt wird, erscheint es sinnvoll, auch das individuelle Durchschauen in dieser Weise subjektiv und aufgabenabhängig zu definieren. Optimal aufgabenangemessene Systemgestaltung zeichnet sich dadurch aus, daß die Zwecksetzungen der Benutzer von den Entwicklern korrekt vorhergesehen wurden.

Überschaut der Benutzer das System nicht nur wie es ist, sondern kann auch Aspekte der Systemgestaltung im eigenen Interesse modifizieren, so *beherrscht* er das System.

Definition: "Beherrschen"

Ein Benutzer *beherrscht* ein Computersystem im Rahmen seiner Arbeit, wenn er es individuell beim Gebrauch durchschaut und die Anwendungs- und Handhabungsfunktionen für seine eigenen Zwecke gebrauchen und anpassen kann.

Ein beherrschbares System bietet dem Benutzer den höchsten Grad an individueller Kontrolle, also nicht nur Orientierung, sondern auch Spielräume zur Beeinflussung, von Freise als externe Voraussetzung für Kontrolle bezeichnet. Unter Beeinflussung wird, wie bereits angesprochen, nicht allein die Möglichkeit verstanden, Systemfunktionen anzustoßen und so das Systemverhalten zu steuern. Der Benutzer soll Umbenennungen vornehmen können, Wege abkürzen durch Zusammenfassen von Funktionssequenzen unter selbstgewählten Namen, er soll nicht-gewünschte Optionen oder Ausgaben ausblenden, Darstellungsformen verändern können. (Vgl. die verschiedenen bei Oberquelle (1994, S. 39 f.) genannten Anpassungstechniken.)

Systemanpassung erfordert immer ein Denken auf einer Metaebene: Nicht der Anwendungsbereich ist der Gegenstand, sondern das Arbeitsmittel Computer. Die Anpassung muß mit Hilfe der geläufigen Interaktionsformen bewerkstelligt werden können. Die Spielräume brauchen nicht so groß zu sein, daß der Benutzer neue Funktionen programmieren oder andere Modifikationen vornehmen kann, für die Programmierkenntnisse erforderlich wären. Das wäre ein zu krasser Wechsel der Betrachtungsebene. Systeme sollen von Benutzern in ihrem Arbeitskontext durchschaut und beherrscht werden. Entsprechend muß auch die Anpassung aus ihrer Perspektive möglich sein.

Der Begriff Beherrschen ist an den Arbeitskontext und die Zwecke der Benutzerin gebunden. Im Sinne einer so definierten Beherrschbarkeit möchte ich auch *Transparenz* definieren.

Definition: "Transparenz"

Ein System ist für eine Benutzerin *transparent*, wenn die Benutzerin sich ein zweckmäßiges mentales Modell vom System gemacht hat (Orientierung) und seine Anwendungs- und Handhabungsfunktionen im Rahmen ihrer Arbeitsaufgabe zielgerichtet einsetzen und einrichten kann (Beeinflussung).

In der Software-Ergonomie wird Transparenz meist als Systemeigenschaft verstanden, die den Aufbau eines mentalen Modells und damit die Orientierung im System erleichtert (vgl. Abschnitt 1). Nach meiner Definition ist Transparenz keine Systemeigenschaft. Sie kann nicht durch Maßnahmen der Systemgestaltung ein für alle Mal und für alle Benutzer und Benutzerinnen hergestellt werden. Transparenz ist ein Idealzustand in der Mensch-Computer-Interaktion, der individuell mehr oder weniger weitgehend erreicht werden kann. Damit wird eine Aussage über die Beziehung zwischen Benutzer/in, Computersystem und Arbeitsaufgabe gemacht.

Alle drei Faktoren wirken zusammen, so daß ein System für individuelle Benutzer im Rahmen ihrer Arbeit zu irgendeinem Zeitpunkt transparent wird. Die Anwendungs- und Handhabungsfunktionalität wird relativ zu einer Arbeitsaufgabe erschlossen und erfahren, die Benutzer entwickeln aufgabenspezifisch Ziele. Von den individuellen Zielen einer Benutzerin hängt es ab, ob bei einem gegebenen Wissensstand von einem individuellen Durchschauen oder Beherrschen gesprochen werden kann. Hat sie einen eingeschränkten Aufgabenbereich, so kann sie das System mit weniger Wissen für ihre Zwecke ausreichend durchschauen und beherrschen. Der Teil der Funktionalität, der Zwecken dient, die eine Benutzerin aktuell nicht verfolgt, muß in ihrem mentalen Systemmodell nicht abgebildet sein.

Es gibt kein objektives Maximum von Transparenz eines Systems für einen Arbeitskontext: Maximale Transparenz kann bedeuten, daß eine Benutzerin tatsächlich alle Anwendungs- und Handhabungsfunktionen kennt, die von den Entwicklerinnen für ihren Arbeitskontext vorgesehen worden waren, und alle vorgesehenen Spielräume nutzen kann. Aber Systeme werden oft in völlig unvorhergesehener Weise genutzt. Ein bekanntes Beispiel hierfür ist die Nutzung von Spreadsheets im Rahmen der Textverarbeitung zum Aufbereiten von textuellen Tabellen, ohne daß irgendetwas gerechnet wird. In diesem Fall wird die verfügbare Funktionalität eines integrierten Bürosystems durchschaut und produktiv zweckentfremdet. Die Benutzerinnen wissen in gewisser Hinsicht noch mehr als die Systementwickler. Sie finden kreativ Spielräume, die von den Entwicklern gar nicht bewußt geschaffen wurden. So wird das System von ihnen auf eine ganz besondere Weise beherrscht.

Meine Transparenz-Definition ist eingebettet in die Systematik des Kontrollkonzeptes und umfaßt bewußt seine beiden Stufen Orientierung und Beeinflussung. Eine Einschränkung aus dem Kontrollkonzept, nämlich daß Kontrolle immer in Relation zu den individuell verfolgten Zielen erlebt wird, findet sich wieder, indem Durchschaubarkeit und Transparenz nur auf den aktuellen Arbeitskontext bezogen werden. Ein transparentes System wird individuell durchschaut, die Benutzerin hat so weit wie nötig Einblick in das System gewonnen und sich ein zweckmäßiges mentales Modell gemacht. Das System steht mit seiner Funktionalität und Handhabung nicht störend, sondern gewissermaßen durchsichtig zwischen der Benutzerin und ihrer Aufgabe. Es ist darüberhinaus ein Stück weit flexibel, indem es sich an die Anforderungen der Benutzer und Benutzerinnen anpassen läßt.

5. Förderung von Transparenz

Oben wurde bewußt der Begriff "beherrschen" gewählt und eingeführt, um die Definition von Transparenz einzuleiten. Der Begriff transportiert besonders gut das Menschenbild, das der menschengerechten Gestaltung von computergestützter Arbeit zugrundegelegt wird, indem er die aktive und kompetente Rolle der Benutzer und Benutzerinnen in der Mensch-Computer-Interaktion betont. Von einer Benutzerin wird Fachkompetenz im Anwendungsbereich und aktives Bemühen um Beherrschung des Systems erwartet. Zunächst muß sie sich ein mentales Modell von Funktionalität und Handhabung machen (durchschauen). Erst darauf aufbauend kann sie das System möglicherweise an eigene Erfordernisse anpassen (beherrschen). Dies wird auf der linken Seite des erweiterten ABC-Modells (Abb. 5) angedeutet.

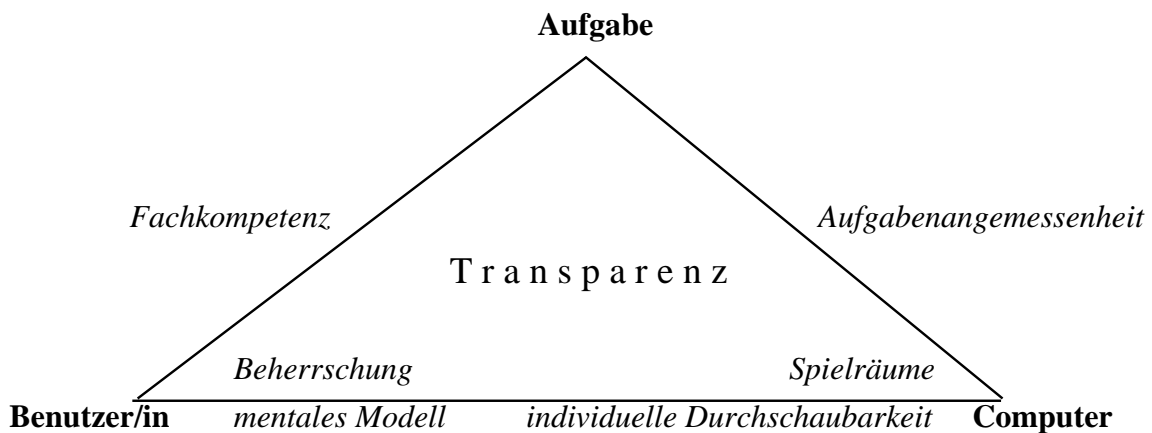


Abb. 5: Transparenz im ABC-Modell

Ein Software-ergonomisch gestaltetes System muß aufgabenangemessen sein (rechte Seite von Abb. 5). Darunter versteht man primär die gute Abstimmung der Anwendungsfunktionalität auf das zu bearbeitende Aufgabenfeld und die möglichst geringe Belastung des Aufgabebearbeitungsvorganges durch handhabungstechnischen Aufwand. Aufgabenangemessenheit charakterisiert die Beziehung zwischen Computersystem und Arbeitsaufgaben. Im Sinne einer Benutzerorientierung wird weiter gefordert, daß das System individuell durchschaubar ist, d.h. die Benutzerin beim Aufbau ihres mentalen Modells vom System unterstützt. Außerdem soll es ihr Spielräume bei der Systemnutzung und zur -anpassung bieten.

Diese Konzepte sind in Abb. 5 entsprechend der Darstellung im Stufenmodell der Kontrolle (Abb. 1) übereinander notiert. Auf der Seite der Benutzerin wird der Aufbau eines mentalen Modells und die Beherrschung des System erwartet; vom Computersystem wird individuelle Durchschaubarkeit und das Angebot von Anpassungsspielräumen gefordert. Auf diese Weise wird Transparenz in der Mensch-Computer-Interaktion ermöglicht. So ist Transparenz im Spannungsfeld zwischen Benutzer/in und Computer angesiedelt, mit dem Aufgabenkontext als wichtiger Randbedingung.

Wie die meisten Forderungen der Software-Ergonomie sind auch Transparenz und Aufgabenangemessenheit nicht unabhängig voneinander. Je aufgabenangemessener ein System gestaltet ist, desto leichter kann sich ein Benutzer, abgestützt auf seine Fachkompetenz, ein Bild davon machen. Umgekehrt belastet ein transparentes System einen Benutzer bei seiner Aufgabenerledigung nicht durch komplizierte Strukturen und Abläufe. Beide Forderungen sind aber nicht identisch: Ein System kann von einem Benutzer durchschaut werden, ohne besonders aufgabenangemessen zu sein. Auch kann es aufgabenangemessen sein, ohne dem Benutzer Anpassungsmöglichkeiten zu bieten, wie sie im Sinne von Transparenz gefordert werden. Somit ist Transparenz auch nicht als *die* allumfassende neue Software-ergonomische Superforderung zu verstehen.

Wie kann nun durch Maßnahmen der Systemgestaltung auf diesen Idealzustand in der Mensch-Computer-Interaktion, den ich Transparenz nenne, hingewirkt werden? Gemäß meiner Definition gibt es zwei Stufen der transparenzförderlichen Systemgestaltung: erstens die Unterstützung beim Aufbau mentaler Modelle und zweitens die Bereitstellung von Spielräumen, auch Anpaßbarkeit oder Individualisierbarkeit genannt.

Das Konzept der Anpaßbarkeit wird erst seit wenigen Jahren intensiver in der Software-Ergonomie diskutiert. (Systematische Überblicke finden sich in Haaks, 1992, und Oppermann, 1994.) Die Forderung nach Anpaßbarkeit drückt gewissermaßen den Respekt vor der Fachkompetenz und den individuell und aufgabenbedingt dynamisch sich wandelnden Anforderungen der Benutzer und Benutzerinnen aus, der in den letzten Jahren entwickelt wurde. Inzwischen werden Systeme speziell auf Anpaßbarkeit hin ausgelegt. Es wird diskutiert, welche programmtechnischen Konzepte Anpaßbarkeit besonders gut zu realisieren erlauben (Haaks, 1992, Oberquelle, 1994). Besondere Konzepte der Benutzungsschnittstelle, die Anpassung ermöglichen, wurden nicht entwickelt. Dies ist auch gar nicht wünschenswert, denn die Benutzer sollten die ihnen vertrauten Interaktionsformen bruchlos zum Zwecke der Systemanpassung nutzen können.

Unter der Überschrift Transparenzförderung werde ich daher im folgenden nur die Unterstützung beim Aufbau mentaler Modelle diskutieren. Beispiele für Maßnahmen dieser Art sind relativ leicht zu finden. Zwei in der Software-Ergonomie sehr populäre Konzepte sollen im folgenden exemplarisch auf ihre Transparenzförderlichkeit untersucht werden. Im Abschnitt 5.1 betrachte ich Systeme mit *Direkter Manipulation*. Sie ermöglichen eine relativ "natürliche" Art der Interaktion. Transparenz entsteht durch Anknüpfen am Vorverständnis der Benutzer. In Abschnitt 5.2 diskutiere ich *Systemkonsistenz*. Konsistente Systeme unterstützen Benutzer durch ihre gleichmäßigen formalen und semantischen Strukturen beim Aufbau neuer Kenntnisse. Erfahrungen mit Teilen des Systems können auf andere Teile verallgemeinert werden. So wird das System schneller erlernbar.

Natürlich gibt es noch andere Maßnahmen bei Systemgestaltung und -einführung, um Be-

nutzerinnen und Benutzer beim Aufbau mentaler Modelle zu unterstützen. Dutke behandelt z.B. die Rolle von Visualisierungen in diesem Zusammenhang (Dutke, 1994, S. 108ff.). Online-Hilfesysteme und Dokumentationen können (bei benutzer- und aufgabenorientierter Gestaltung) nützlich sein. Gezieltes Training, sorgfältig geplante Einführung und intensive Betreuung bei der Nutzung von Systemen sowie die explizite Ermunterung zur gemeinsamen Einarbeitung und zum fortwährenden informellen Erfahrungsaustausch unter Kollegen und Kolleginnen sind unverzichtbare Randbedingungen für die Ausbildung von Kompetenzen und letztendlich das Erreichen von Transparenz.

5.1 Transparenz durch Direkte Manipulation?

Direkte Manipulation (DM) ist eine spezielle Interaktionsart mit dem Computer, bei der die Arbeitsgegenstände und die darauf möglichen Aktionen permanent und möglichst anschaulich graphisch auf dem Schirm dargestellt werden. Diese Modellwelt kann der Benutzer räumlich manipulieren: Er zeigt auf die Objekte und bewegt sie, er wählt die Aktionen aus "Menüs" oder über "Knöpfe" auf dem Schirm aus. Er führt also einfache körperliche Aktionen aus und muß sich keiner komplexen sprachlichen Syntax bedienen. Das System erlaubt ihm schnelle, inkrementelle und rücknehmbare Operationen, deren Wirkung auf den Arbeitsgegenstand sofort sichtbar gemacht wird. (Vgl. auch Shneiderman, 1983.)

Die graphische Darstellung einer Modellwelt auf dem Schirm hat für den Benutzer den Vorteil, daß er sich die Systemelemente und Funktionen nicht alle zu merken braucht. Er muß anhand der graphischen Symbolik und der sprachlichen Benennung Objekte und Aktionen (wieder-)erkennen und anwählen. Hierbei hilft ihm der starke Einsatz von visuellen und sprachlichen Metaphern.

Metaphern ziehen Analogien zwischen einem vertrauten und einem unbekanntem Bereich (Quelle bzw. Ziel). Sie ermöglichen damit eine Übertragung vorhandenen Wissens in neue Kontexte und erleichtern so die Orientierung im neu zu erschließenden Bereich: Metaphern reduzieren die Komplexität des neuen Bereiches zunächst auf bekannte Konzepte, von denen ausgehend dann neues Wissen erworben werden kann. Der Aufbau mentaler Modelle kann durch Einsatz von Metaphern besonders gut unterstützt werden, denn mentale Modelle beruhen zu einem beträchtlichen Teil auf Analogiebeziehungen. (Genauer vgl. Dutke, 1994.)

Metaphern sind nicht wahr oder falsch, sondern sie variieren in ihrer Erkennbarkeit und Nützlichkeit. Die Analogie zwischen Quell- und Zielbereich muß stark genug sein, damit die Metapher erkannt werden kann. Sie darf andererseits nicht zu stark sein, weil die Metapher dann Gefahr läuft, nicht als solche erkannt, sondern wörtlich interpretiert zu werden. Nützlich ist eine Metapher, wenn die Menge an übertragbaren Relationen groß genug ist.

Aufgrund ihrer grundsätzlichen Unbestimmtheit haben Metaphern eine motivierende Wirkung im Lernprozeß. Der Hörer muß die Metapher erkennen und aktiv die darin nahegelegten Analogieschlüsse ziehen. Das Wissen, daß es Eigenschaften des Quellbereichs gibt, die sich im Zielbereich nicht wiederfinden, erregt seine Neugier und regt ihn zu weiterem Lernen an. "Die partielle Nichtpassung zwischen Basis- und Zielbereich ist notwendiger Bestandteil einer Metapher", schreibt Dutke (1994, S.22).

Es kann sinnvoll sein, auf die Grenzen einer Metapher explizit hinzuweisen, falls eine Überinterpretation für die Benutzerin zu unliebsamen Überraschungen führen kann. So kann z.B. die sog. Zwischenablage (engl. Clipboard) beim Apple Macintosh nur einen Eintrag zur Zeit enthalten. Wird der Zwischenablage ein neuer Eintrag zugewiesen, so wird dieser nicht "neben" die vorhandenen Einträge "gelegt" oder "gehängt", sondern der vorhandene Eintrag wird ersetzt und geht verloren; nicht einmal im Systempapierkorb ist er danach wiederzufinden. Dieser Effekt hat damit zu tun, daß die Zwischenablage aus Systemsicht kein Behälter (Ordner), sondern ein spezielles, permanent vorhandenes Dokument ist. Für die Benutzerin prägt die Bezeichnung Zwischenablage jedoch eher die Erwartung eines Behälters, der mehrere Einträge enthalten kann. Wird sie nicht auf die ein-

geschränkte Gültigkeit der Metapher hingewiesen, so läuft sie Gefahr, bei der Arbeit Dokumententeile zu verlieren.

Bei Systemen mit vielfältiger Funktionalität reicht meist eine einzelne Metapher nicht aus. Bei der Verwendung mehrerer Metaphern ist darauf zu achten, daß alle verwendeten Metaphern einigermaßen zueinander passen und keinen extrem widersprüchlichen Umgang nahelegen. Dadurch würde der Benutzer verunsichert und der positive Effekt der Metaphern ginge verloren. Optimal ist eine Reihe von hierarchisch geschachtelten Metaphern, wie etwa beim Apple Macintosh realisiert: Die generelle Handhabung wird als "Direkte Manipulation" bezeichnet; der Bildschirm wird als "Schreibtisch" beschrieben, dort erwartet man bestimmte Objekte wie z.B. Dokumente, Kopien, Ordner. Diese findet man im elektronischen Schreibtisch tatsächlich vor. Sie lassen Aktionsmöglichkeiten erwarten wie Öffnen, Lesen, Kopieren, (partiell) Weiterverwenden. Natürlich kann eine solche Analogie zwischen gegenständlicher und Modell-Welt nicht durchgehalten werden, weil damit auch die Vorteile der Verlagerung von Arbeit auf den Computer verloren gingen. Ergänzend zur Schreibtisch-Metapher und all ihren Untermetaphern verwendet der Macintosh andere Metaphern wie z.B. Menü, Fenster, Rollbalken, die nicht aus dem Bürobereich stammen. Sie alle stehen aber unter dem gemeinsamen Dach der Direkten Manipulation und behindern sich gegenseitig nicht.

Besonders in frühen Lernstadien, wo vereinfachte Darstellungen eine grobe Orientierung ermöglichen, sind Metaphern lernförderlich. Später, wenn das übertragene Wissen fest mit dem neuen Bereich assoziiert wird, dienen sie nur noch als Erinnerungshilfen. Etwa mit dem Konzept "Ordner" wird nach einiger Zeit nicht mehr die enge Vorstellung eines festen Einbandes zum Abheften von Papieren verbunden. Ein Ordner wird genereller zum Behälter, der sowohl Dokumente als auch weitere Ordner enthalten kann, obwohl das den Bedingungen in der realen Welt nicht mehr entspricht. Auch das Wort "Menü" wird metaphorisch verwendet. Korrekt übersetzt heißt das englische Wort "menu" im Deutschen soviel wie Speisekarte. Hier soll die Assoziation genutzt werden, daß die Leserin sich eine oder mehrere Einträge aus einem Angebot auswählen kann. Trotz der inkorrekten Übersetzung mit "Menü" (im Deutschen bezeichnet dies nicht die Karte, sondern eine festgelegte Speisenfolge) ist die Metapher auch im Deutschen noch erkennbar und nützlich und hat sich bekanntlich durchgesetzt. Und ein letztes Beispiel: Die "Papierkorb"-Metapher im Zusammenhang mit dem Wegwerfen nicht mehr benötigter Objekte leuchtet jedem Anfänger sofort ein. Daß man aber ein Diskettensymbol auf den Papierkorb ziehen muß, um die Diskette beschleunigt auswerfen zu lassen, widerspricht dieser anfänglich vorgestellten Wirkung so sehr, daß jeder Anfänger zögert, diese Aktion zum ersten Mal auszuführen. Danach merkt er sich leicht diese Besonderheit, ohne daß die Metapher dadurch für ihn gleich wertlos würde.

Auch der Begriff "Direkte Manipulation" selbst ist eine Metapher. Im Gegensatz zur sog. Kommando-Interaktion, die ausschließlich auf der sprachlichen Ebene läuft, legt er einen quasi-gegenständlichen, unvermittelten Umgang mit dem System nahe, der jedem aus dem täglichen Leben vertraut ist. Damit fordert er zum Ausprobieren solcher vertrauter Handlungen im neuen Kontext heraus. Direkte Manipulation ist ein gutes Beispiel für einen gewünschten Nebeneffekt von Metaphern: Sie sollen eine positive emotionale Wirkung haben. Der Begriff Direkte Manipulation (im Deutschen wäre eigentlich "Direkte Handhabung" noch besser) beugt Berührungsängsten gegenüber dem Computer vor, denn zeigen, greifen, führen, Knöpfe drücken u.ä. kann eigentlich jeder.

Am Beispiel der Metapher DM kann man auch studieren, was passiert, wenn eine Metapher besonders gut zu passen scheint: ihre Grenzen werden nicht mehr gesehen. Die Metapher wird nicht mehr als solche erkannt, sondern wörtlich interpretiert. Dieses Mißverständnis führt dazu, daß Benutzer von DM-Systemen überrascht und ungehalten reagieren, wenn die Metapher in ihren Augen stellenweise nicht trägt, z.B. wenn man "Lineale" nicht "ziehen" kann (s.u.) oder wenn Datei-Ikone nicht genau dort landen, wohin man sie bewegt hat. Die Art der direkten Handhabung von Objekten am Schirm wird so vorrangig, daß mancher Benutzer, der seine Diskette auswerfen lassen möchte und keine intuitive "di-

rekte" Lösung findet, gar nicht darauf kommt, in den Systemmenüs nach einem entsprechenden Kommando zu suchen.

Oft lassen sich bekannte Objekte und Handlungen aus dem Anwendungsbereich auf der Benutzungsoberfläche in naheliegender Weise symbolisch darstellen und benennen. Eigentlich kann es aber gar nicht im Sinne einer Computerunterstützung sein, die gegenständliche Welt perfekt nachzubilden. Durch die Verlagerung auf den Computer ergeben sich neue Möglichkeiten der Verarbeitung. Für diese neuen Operationen ist es schwieriger, geeignete Bilder zu finden. Das integrierte Bürosystem RagTime z.B. nutzt im Zusammenhang mit Formatieroperationen die "Lineal"-Metapher. Lineale werden z.T. systemseitig im Text gesetzt, können aber auch vom Benutzer hinzugefügt werden. Lineale sind nicht immer sichtbar, können aber jederzeit sichtbar gemacht werden. Sie liegen dann als dünne Balken im Text, tragen regelmäßige Einteilungen und können Tabulatoren aufnehmen und anzeigen. Aus der Sicht der Systementwicklerinnen ist die Lineal-Metapher bereits ein Zugeständnis an die Denkweise von Benutzern.

RagTime erlaubt es, Formatierungsoperationen auf explizit umgrenzte Textbereiche anzuwenden. Der eigentliche Sinn von Linealen in RagTime, nämlich diese Umgrenzung, ist nur noch schwach mit den bekannten Eigenschaften von Linealen in Verbindung zu bringen. So ist es für Benutzer, auch wenn sie von Linealen in RagTime wissen, meist nicht intuitiv klar, daß sie z.B. zur Änderung der Zeilenabstände in einem Textstück den Cursor im betreffenden Gebiet positionieren und dann in das "Lineale"-Menü gehen müssen, um dort die entsprechenden Einstellungen vornehmen zu können. Die Lineal-Metapher ist offensichtlich nicht sehr nützlich, weil es kaum wichtige Eigenschaften des Zielbereiches gibt, die aus Eigenschaften des Quellbereiches abgeleitet werden können.

Das äußere Erscheinungsbild von Linealen verführt Benutzer dazu zu versuchen, Lineale bei Bedarf an andere Stellen zu "ziehen". Diese prototypische Operation der Direkten Manipulation läßt sich in RagTime aber nicht auf Lineale anwenden. In den Augen der Benutzer ist ein Lineal ein graphisches Element, und diese können üblicherweise verschoben werden. Aus Systemsicht sind Lineale jedoch Textelemente, die wie andere Textelemente nur durch Löschen und Einfügen an eine andere Stelle versetzt werden können. Insbesondere werden sie mitgelöscht, wenn das Zeichen gelöscht wird, an dem sie "kleben". Hier bricht die Lineal-Metapher also ganz zusammen, soweit sie überhaupt getragen hat: Ein Lineal ist nur eine Markierung im Text.

Eine herausragende Eigenschaft von DM-Systemen ist ihr intensiver Einsatz von anschaulichen Metaphern. Damit wird der Benutzer beim Erschließen des Systems, d.h. beim schrittweisen Aufbau seines mentalen Modells unterstützt. Hierbei ist hilfreich, daß das System abhängig vom gerade angewählten Objekt auf dem Schirm die jeweils aktuell möglichen Aktionen auf Menüleisten, in pulldown-Menüs o.ä. anzeigt. Die im Verlaufe seiner Aufgabenbearbeitung möglichen Aktionen werden dem Benutzer also immer wieder vor Augen geführt. So hat er Gelegenheit, besonders die für seinen individuellen Arbeitskontext relevanten Konzepte zu lernen und ein zweckmäßiges mentales Modell zu entwickeln.

DM-Systeme ermutigen zu explorierendem Handeln, denn sie geben dem Benutzer schnelles Feedback auf seine Aktionen, und die angestoßenen Operationen sind unmittelbar nach ihrer Ausführung ohne Schaden rücknehmbar. Ausgehend von einem zunächst bruchstückhaften mentalen Modell und ersten Hypothesen über das Systemverhalten spielt der Benutzer mögliche Aktionen in Gedanken durch. Die Güte seiner geistigen Simulation kann er dann gefahrlos durch Probehandeln am System überprüfen. Er sieht, was passiert, findet sein mentales Modell bestätigt oder kann es schrittweise modifizieren und erweitern. So wird seine aktive Suche nach Orientierung unterstützt, das System wird für ihn relativ schnell überschaubar.

Besonders wichtig ist bei DM-Systemen eine gute visuelle Darstellung am Bildschirm. Nur so können bildlich repräsentierte Metaphern erkannt und Systemfeedback vom Benutzer interpretiert werden. Ich möchte zum Themenkomplex Visualisierung an dieser Stelle

keine Ausführungen machen und verweise stattdessen z.B. auf Dutke (1994).

Ich fasse zusammen: Systeme mit Direkter Manipulation schlagen durch ihre Metaphorik Brücken zwischen der Gedankenwelt der Benutzerinnen und Benutzer und den Objekten und Funktionen des Systems. Sie verringern die "Distanz", wie Hutchins, Hollan und Norman (1986) schreiben. DM-Systeme ermöglichen die Systemexploration während der Aufgabenbearbeitung, erleichtern so den Erwerb aufgabenrelevanter Konzepte. Auf diese Weise fördern sie das individuelle Durchschauen beim Systemgebrauch. Sofern Möglichkeiten zur individuellen Anpassung des Systems gegeben sind, sind sie ebenfalls in Direkter Manipulation möglich.

5.2 Transparenz durch Konsistenz?

Konsistenz ist eine zentrale Forderung im Sinne benutzergerechter Systemgestaltung. *Konsistente Systeme* verhalten sich regelhaft, gleichmäßig, der Benutzer kann Erwartungen ausbilden, das System reagiert erwartungskonform. Konsistenz ist das "Prinzip der geringsten Überraschung". Es wird angenommen, daß konsistente Systeme aufgrund ihrer Struktur leichter zu erlernen sind.

Allgemein bedeutet Konsistenz soviel wie strukturelle Übereinstimmung, Widerspruchsfreiheit; es wird also eine Beziehung charakterisiert. Je nachdem was zueinander in Beziehung gesetzt wird, ergeben sich sehr unterschiedliche Bedeutungen für die Forderung nach konsistenten Systemen.

Interne Konsistenz heißt, daß in verschiedenen Systemteilen z.B. das Layout von Bildschirmmasken, die Bedeutung von Funktionstasten, die Interaktionsart, der Aufbau von Menüs oder die Anwendbarkeit von Funktionen auf Objekte gleichartig gestaltet sind. Ist dem Benutzer diese Regelmäßigkeit erst klargeworden, so spart er dadurch Lernaufwand ein. Die Erfahrungen, die er mit einem Systemteil macht, bekommen exemplarischen Charakter: Er kann sie auf andere Teile übertragen und muß sich das System nicht an jeder Stelle von vorne erschließen.

Externe Konsistenz bedeutet dagegen, daß verschiedene Systeme bezüglich eines oder mehrerer dieser Aspekte gleichartig gestaltet sind. So wird dem Benutzer das Umsteigen zwischen Systemen erleichtert, denn er kann beim Aufbau seines mentalen Modells vom neuen System von Vorstellungen über das alte System ausgehen.

Ein System kann aber auch (extern) konsistent mit der Anwendungsumgebung sein, die mit ihren Begriffen, Strukturen und Abläufen das Denken der Benutzer prägt: Sind Funktionen und Objekte in für den Benutzer vertrauter und naheliegender Weise benannt? Ist die Gesamtfunktionalität dem Anwendungsbereich angemessen, und ist sie intern angemessen gegliedert? Gestattet das System ein Beibehalten gewohnter und bewährter Arbeitsabläufe? Dann kann sich der Benutzer das System ausgehend von seinem Wissen über den Anwendungsbereich erschließen. In dieser zweiten Spielart externer Konsistenz verliert Konsistenz den Charakter einer rein formalen Systemeigenschaft, denn es wird Bezug auf individuelle Denkweisen in verschiedenen Arbeitskontexten genommen. Es ergibt sich eine Überschneidung mit dem Begriff Aufgabenangemessenheit.

Interviews mit Systementwicklern ergaben, daß Konsistenz keine wohldefinierte Forderung ist und daß - über die intern/extern-Differenzierung hinausgehend und gewissermaßen quer dazu - mit Konsistenz im Detail sehr verschiedene Vorstellungen verbunden wurden (Maaß, Rosson & Kellogg, 1987). In Beispielen wurden in erster Linie physikalische und syntaktische Aspekte der Interaktion angeführt: Art der Geräte, visuelle Darstellung und Struktur der Ausgabe, syntaktische Form der Eingaben.

Kellogg (1987) bezeichnet Konsistenz auf dieser Ebene als "procedural consistency", also Verfahrenskonsistenz oder *prozedurale Konsistenz*. Die heute von Herstellern und Anwendern eingesetzten Richtlinien und Regelwerke ("styleguides") für graphische Oberflächen zielen ausschließlich auf diese Art der Konsistenz (vgl. Nielsen, 1989). Die meist

dazugehörigen Schnittstellen-Entwicklungswerkzeuge helfen bei der internen und systemübergreifenden Realisierung. Prozedurale Konsistenz kann sich nachhaltig auf die Performanz der Benutzer und Benutzerinnen am System auswirken. Wenn sie sich allerdings auf eine ästhetische Glättung der Systemoberfläche beschränkt, hat sie diesen Effekt nicht: Piktogramme, die einem einheitlichen graphischen Stil folgen, ergeben vielleicht ein angenehmeres Gesamtbild, sind aber dadurch nicht etwa leichter verständlich. Es ist also nicht sinnvoll, bei einer formalen Konsistenz stehenzubleiben. Zur gleichmäßigen Gestaltung muß immer auch die inhaltlich angemessene Gestaltung kommen.

Entsprechend bezieht sich die begrifflich-gedankliche oder *konzeptuelle Konsistenz* ("conceptual consistency"), die Kellogg (1987) gegen die prozedurale absetzt, auf semantische Konzepte. Sie kennzeichnet einerseits die Abbildung von Arbeitszielen der Benutzerinnen auf Systemobjekte und -funktionen und andererseits die interne Kohärenz der Systemstruktur. Am Beispiel verschiedenartiger Zerlegungen von Aufgaben in Folgen von Funktionsaufrufen im System zeigt Kellogg, daß prozedural *und* konzeptuell konsistente Systeme schneller gemeistert wurden als nur prozedural konsistente Systeme: Vorgehensweisen wurden besser erinnert bzw. richtig abgeleitet, Aufgaben wurden schneller erledigt, und die Benutzerinnen gingen selbstsicherer mit dem System um. Sie überschauten das System offensichtlich schneller und konnten es in gewünschter Weise steuern. Die konzeptuelle Konsistenz wirkt also neben der prozeduralen Konsistenz eindeutig positiv im Sinne von Systemtransparenz.

Sicher ist, daß nur solche Konsistenz Sinn macht, die vom Benutzer auch wahrgenommen wird. Nur dann erleichtert sie ihm den Aufbau seines mentalen Systemmodells. Eine von den Systementwicklern angelegte Regelmäßigkeit kann Benutzern gänzlich verborgen bleiben oder sogar verwirrend wirken, wenn sie mit Kategorien arbeitet, die den Benutzern nicht vertraut oder nicht wichtig sind. Wenn z.B. alle Textdateien, in die auch Videoclips eingebunden sind, automatisch in einem besonderen Verzeichnis abgelegt werden und solche ohne Videos in einem anderen, so mag das aus Entwicklersicht eine geplante und sinnvolle Regelmäßigkeit sein. Für Benutzer, die aus ihrem Arbeitskontext heraus mal mit und mal ohne Videoclips arbeiten, ist diese Unterscheidung möglicherweise völlig unwichtig und die getrennte Archivierung verwirrend. (Wer einmal im Großraumabteil des neuen ICE-Zuges systematisch nach seinem reservierten Sitzplatz gesucht hat, kennt eine ähnliche Verwirrung.)

Ist ein System durchgängig formal konsistent gestaltet, so kann es vollständig formal beschrieben werden. Dies ist aus der Sicht der Systementwicklerinnen wünschenswert. Auch kompetente Benutzer können sich auf diese Weise ein vollständiges Bild vom System machen. Diese formale Konsistenz ist eine reine Systemeigenschaft, die auch ohne Einbezug von Benutzern sicherzustellen ist. Zum Erreichen von Transparenz im oben definierten subjekt- und aufgabenbezogenen Sinne ist eine vollständige formale Konsistenz allerdings nicht nötig.

Auch Grudin betont in seinem Artikel "The Case Against Consistency" (Grudin, 1989) das "Primat der Aufgabe" und macht deutlich, daß eine globale Konsistenz unsinnig ist, wenn sie nicht dem Gedanken der Aufgabenangemessenheit untergeordnet ist. In erster Linie müssen die Systemfunktionen und die Art der Handhabung die verschiedenen Aufgabenabläufe der Benutzer unterstützen; in einem zweiten Schritt ist dann eine konsistente Gestaltung anzustreben. Bei diesem Vorgehen können sich interne und externe Inkonsistenzen ergeben. Aber der Aufbau des mentalen Modells beim Benutzer wird dadurch nicht beeinträchtigt, denn Benutzer erschließen sich Systeme meist ausgehend von konkreten Aufgaben. In Textverarbeitungsprogrammen des Macintosh z.B. gibt es für viele, aber nicht alle Menü-Einträge Kommandokürzel. Welche Option abkürzbar ist, wurde von den Entwicklern abhängig von der Art der damit verbundenen Operation und der absehbaren Häufigkeit ihrer Benutzung bei Textverarbeitungsaufgaben festgelegt. Die Entscheidung hat also inhaltliche Gründe. Die Abkürzungen selbst haben eine konsistente, leicht erinnerbare Form ("Apfel"+ Zeichen). Wenn sich diese Lösung für die Aufgabenbearbeitung lokal als praktisch erweist (schnellerer Kommandoaufruf), wird den Benutzer eine damit verbunde-

ne (globale) Inkonsistenz (nicht alle Kommandos sind abkürzbar) nicht stören.

Ich fasse zusammen: Interne und externe Konsistenz erleichtern den Benutzern und Benutzerinnen den Aufbau mentaler Systemmodelle und tragen so zur Transparenz in der Mensch-Computer-Interaktion bei. Allerdings darf sich die Konsistenz nicht auf rein formale, systeminterne Aspekte beschränken. Erst wenn konsistente Gestaltung Bezug nimmt auf individuelle und aufgaben-inhaltliche Konzepte, wird sie positiv wirksam. Eine Konsistenzforderung ist also nur sinnvoll, wenn Konsistenz breit als ein Konzept verstanden wird, das - genau wie Transparenz - die Beziehung zwischen dem Benutzer, seiner Aufgabe und dem System charakterisiert. Eine derartige Kombination von konzeptueller und prozeduraler Konsistenz führt, falls sie so angelegt sind, daß die Benutzer und Benutzerinnen sie wahrnehmen können, zum besten Ergebnis. Formale Konsistenz ist nur sinnvoll auf der Basis aufgabenorientierter Gestaltungsvorentscheidungen.

Die konsistente Gestaltung muß sich auch auf etwaige Anpassungsfunktionen erstrecken. Konsistenz an sich trägt allerdings nichts zur Erweiterung von Spielräumen für die Benutzer bei.

6. Zusammenfassung

Die Forderung nach Systemtransparenz kann zu den Standardforderungen der Software-Ergonomie gezählt werden. Sie ist ähnlich allgemein und wenig operational wie die einschlägigen DIN- und ISO-Prinzipien und eng mit ihnen verwandt. Allerdings ist nicht immer ganz klar, was mit Transparenz genau gefordert wird, denn Definitionen werden nur selten gegeben. Meist wird die Bedeutung darauf reduziert, daß die Benutzer Einblick in die inneren Vorgänge im System erhalten sollen. Transparenz wird als Systemeigenschaft charakterisiert.

Nach meiner Definition ist Transparenz keine Systemeigenschaft, sondern ein Idealzustand in der Mensch-Computer-Interaktion. Transparenz ist ein subjektives Konzept. Die Definition lehnt sich an das Kontrollkonzept an, das als zentral für menschengerechte Arbeitsgestaltung gilt. Ein System ist für eine Benutzerin im Rahmen ihrer Arbeitsaufgaben transparent, wenn sie sich ein zweckmäßiges Modell davon gemacht hat und Spielräume nutzen kann, um das System für ihre Zwecke anzupassen. Beide Aspekte von Kontrolle, sowohl Orientierung als auch Beeinflussung, sind im Falle von Transparenz gegeben.

Was ist mit dieser Begriffsklärung gewonnen? Es wird deutlich, in welche Richtung die Gestaltungsbemühungen gehen müssen und wohin nicht: Es wurde die Differenzierung zwischen grundsätzlicher Durchschaubarkeit im Entwicklungskontext und individueller Durchschaubarkeit beim Gebrauch getroffen. Systementwickler und -entwicklerinnen durchschauen ihre eigenen Systeme oder Moduln auf programmtechnischer Ebene. Bei größeren, von mehreren Entwicklern oder Entwicklergruppen entwickelten Systemen kann meist nur von einem "kollektiven Durchschauen" gesprochen werden: Gemeinsam können die Entwicklerinnen und Entwickler das System noch durchschauen. Die einzelnen Entwickler durchschauen das Gesamtsystem aber nicht mehr im Detail. Es ist nicht sinnvoll, Benutzern auf dieser Ebene Einblick zu verschaffen.

Benutzer betrachten Systeme aus einer anderen Perspektive, nämlich der ihres fachlichen Problems. Um selbständig und kompetent mit einem System arbeiten können, brauchen sie nicht das gesamte System zu durchschauen, sondern nur die aufgabenrelevanten Teile. Dies wurde als individuelles Durchschauen beim Gebrauch bezeichnet. Transparenzförderliche Systemgestaltung muß versuchen, dem Benutzer den Aufbau eines mentalen Modells zu erleichtern, das zweckmäßig für seine Aufgaben und Ziele ist. Das mentale Modell ist die interne Voraussetzung für weitergehende Kontrolle, die externe Voraussetzung ist, daß das System dem Benutzer auch Spielräume bietet, innerhalb derer er es an seine speziellen Anforderungen anpassen kann.

Im Kontext der Software-Ergonomie gehört Transparenz zu den Forderungen, die im Sinne von Qualifizierung und Persönlichkeitsförderung gestellt werden. Auch die von mir ge-

gebene Definition, die auf angemessene Kontrolle für die Benutzer hinausläuft, läßt sich in dieser Weise einordnen. Da das menschliche Kontrollbedürfnis aber abhängig von individuellen Zielen ist, schränke ich die in der Software-Ergonomie gängige Transparenzforderung ein: Benutzer sollen nicht alles, sondern nur das im Rahmen ihrer Arbeit Relevante durchschauen. Auf der anderen Seite fordere ich mehr: Die Benutzer sollen Systeme nicht nur durchschauen, sondern auch beeinflussen können. Fachkompetenz auf Seiten der Benutzer und aufgabenangemessene Gestaltung des Systems machen es wahrscheinlicher, daß Transparenz in der Mensch-Computer-Interaktion erreicht wird.

Auch die eingangs aufgebrachte Frage, ob Benutzer Transparenz "wünschten" oder nicht, läßt sich nun differenzierter diskutieren. Computerbenutzer sind Fachkräfte, die an der Erledigung von Aufgaben arbeiten. Hierbei benutzen sie Computer als Hilfsmittel. Die in der Informatik übliche Bezeichnung "Benutzer" definiert die Rolle dieser Menschen nur noch in Relation zum Computer (vgl. Grudin, 90). Dies spiegelt die Perspektive der Systementwickler wider. Die Fachkräfte selbst sehen sich nicht primär als Benutzerinnen und Benutzer von Computern, sondern als KassiererIn, Schornsteinfeger, Versicherungssachbearbeiter, Ärztin ... Die meisten Benutzer möchten nicht ihre fachliche Perspektive aufgeben und ohne Not mit programmiertechnischen Konzepten konfrontiert werden. Im Kontext ihrer individuellen Aufgabenstellung wollen sie die relevanten Systemteile durchschauen und beherrschen. Sie brauchen Hilfe beim Aufbau zweckmäßiger mentaler Modelle und Spielräume zur individuellen Anpassung des Systems. Damit werden ihre Handlungsspielräume erweitert, und ihre Qualifizierung wird gefördert. Natürlich ist es jeder Benutzerin freigestellt, sich mit beliebig vielen Systemaspekten vertraut zu machen. Es darf aber nicht aufgrund von ungeschickter Systemgestaltung von ihr verlangt werden.

Transparenz in der Mensch-Computer-Interaktion als Gestaltungsziel bedeutet nicht, daß jegliche Systemkomplexität zu vermeiden ist. Aufgabenangemessene Systemgestaltung spiegelt notwendig immer die Komplexität des Anwendungsbereiches und der Aufgaben wider und nutzt die Fachkompetenz der Benutzer und Benutzerinnen. Was zu vermeiden ist, ist eine aus der Sicht des Benutzers unnötige Komplexität (Kompliziertheit), die er bei seiner Aufgabenbewältigung nicht benötigt, sondern die ihn behindert.

Meine Ausführungen haben sich auf die Situation computergestützter Arbeit bezogen. Hauptanliegen der Computerbenutzerinnen und -benutzer ist dabei die Bewältigung ihrer Arbeitsaufgaben. Die Technik soll sie dabei unterstützen und ihren Arbeitsfluß nicht stören. Arbeitszufriedenheit und Motivation nähren sich aus verschiedenen Quellen. Gut gestaltete technische Unterstützung ist nur eine davon. Das Gefühl, ein Computersystem individuell zu durchschauen und zu beherrschen, kann Spaß machen. Hierfür steht das Konzept der Transparenz. Vorrangig ist aber immer die menschengerechte Gestaltung der Aufgaben selbst.

In anderen Zusammenhängen, etwa bei Computerspielen, gelten ganz andere Randbedingungen. Hauptanliegen beim Spiel ist die Bewältigung von Komplexität, Spaß entsteht u.a. durch das Gefühl der Herausforderung. Anfänglich müssen die Spielregeln durchschaut werden, und die Situation auf dem Schirm muß interpretierbar sein. Hat der Benutzer im Spiel einen Zustand von Transparenz erreicht, so führt dies meist umgehend zum systemseitigen Anheben der Komplexität und damit zu erneuter Intransparenz. Mehr Spielräume erlauben mehr Kontrollmöglichkeiten und geben damit auch mehr Gelegenheit zum Scheitern: Dieser Wechsel zwischen Intransparenz und Transparenz macht gerade den Reiz vieler Computerspiele aus.

Auch Systementwickler empfinden ihre Arbeit am Computer häufig als Spiel, als Bewältigung einer Herausforderung. Bei der Systementwicklung darf man nicht den Fehler begehen, von den Benutzern dieselbe Einstellung zu erwarten. Transparenz und Intransparenz sind subjektive Konzepte. Menschengerechte Systemgestaltung muß die Voraussetzungen für Transparenz in der Mensch-Computer-Interaktion gewährleisten.

Software-ergonomische Kriterien sind subjektiv und gelten kontextabhängig. Eine Einschätzung von Systemqualität anhand der Kriterien kann nur durch Benutzerinnen oder mit

ihrer Hilfe geschehen. Systementwickler können nicht anstelle von Benutzerinnen entscheiden, ob ein System selbstbeschreibend ist, wann es erwartungskonform, wann ausreichend übersichtlich oder steuerbar ist. Dazu sind die Perspektiven der Systementwickler und -benutzerinnen viel zu unterschiedlich. Transparenz erweist sich somit als typische Software-ergonomische Forderung.

Dank

Für Diskussionen und Anregungen danke ich meinen Kollegen Ralf Klischewski, Horst Oberquelle und Peter Scheffé sowie den studentischen Teilnehmerinnen und Teilnehmern meines Seminars "Systemtransparenz und -konsistenz" im WS 93/94.

7. Literatur

Biedenkopf, 1994

Kurt H. Biedenkopf. Komplexität und Kompliziertheit. Informatik-Spektrum, 17, 2, 1994, S. 82-86.

Bødker, 1991

Susanne Bødker. Through the Interface. A Human Activity Approach to User Interface Design. Laurence Erlbaum, Hillsdale, N.J., 1991.

DIN 66234, Teil 8

DIN 66234, Teil 8. Bildschirmarbeitsplätze. Grundsätze ergonomischer Dialoggestaltung. Beuth, Köln, 1988.

Dutke, 1994

Stephan Dutke. Mentale Modelle: Konstrukte des Wissens und Verstehens. Kognitionspsychologische Grundlagen für die Software-Ergonomie. Verlag für Angewandte Psychologie, Göttingen, Stuttgart, 1994.

Fitter, 1979

Mike Fitter. Towards More "Natural" Interactive Systems. International Journal of Man-Machine Studies, 11, S. 339-350.

Frese, 1987

Michael Frese. A Theory of Control and Complexity: Implications for Software-Design and Integration of Computer Systems into the Work Place. In: Michael Frese, Eberhard Ulich, Wolfgang Dzida (Hrsg.). Psychological Issues of Human-Computer Interaction in the Work Place. Elsevier Science (North Holland), Amsterdam, 1987, S. 313-337.

Frese & Brodbeck, 1989

Michael Frese, Felix C. Brodbeck. Computer in Büro und Verwaltung. Psychologisches Wissen für die Praxis. Springer, Berlin u.a., 1989.

Grudin, 1989

Jonathan Grudin. The Case Against User Interface Consistency. Communications of the ACM, 32, 10, 1989, S. 1164-1173.

Grudin, 1990

Jonathan Grudin. Interface. In: Proc. Conf. on Computer-Supported Cooperative Work, CSCW'90, Los Angeles, 1990, S. 269-278.

Haaks, 1992

Detlef Haaks. Anpaßbare Informationssysteme. Verlag für Angewandte Psychologie, Göttingen, Stuttgart, 1992.

Heidegger, 1984

Martin Heidegger. Sein und Zeit. Niemeyer, Tübingen, 1984. (Original 1927)

Herrmann, 1994

Thomas Herrmann. Grundsätze ergonomischer Gestaltung von Groupware. In: Anja Hartmann, Thomas Herrmann, Markus Rohde, Volker Wulf (Hrsg.). Menschengerechte Groupware - Software-ergonomische Gestaltung und partizipative Umsetzung. Teubner, Stuttgart, 1994, S. 65-107.

Holtzblatt, Jones & Good, 1988

Karen Holtzblatt, Sandra Jones, Michael Good. Articulating the Experience of Transparency: An Example of Field Research Techniques. SIGCHI Bulletin, 20, 2, 1988, S. 45-47

Hutchins, Hollan & Norman, 1986

Edwin L. Hutchins, James D. Hollan, Donald A. Norman. Direct Manipulation Interfaces. In: Donald A. Norman, Stephen W. Draper (Hrsg.). User Centered System Design. New Perspectives on Human-Computer Interaction. Lawrence Erlbaum, Hillsdale, N.J., 1986, S. 87-124.

ISO 9241, Part 10

Ergonomic Requirements for Office Work with Visual Display Terminals (VDTs), Part 10: Dialogue Principles, Draft International Standard, Februar 1993.

Keil-Slawik & Holl, 1987

Reinhard Keil-Slawik, Friedrich Holl. Transparenz von Dialogsystemen. Abschlußbericht über die gleichnamige Arbeitsgruppe, Tagung Mensch-Maschine-Kommunikation, Peiting, (nicht veröffentlicht), 1987.

Kellogg, 1987

Wendy A. Kellogg. Conceptual Consistency in the User Interface. Effects on User Performance. In: Hans-Jörg Bullinger, Brian Shackel (Hrsg.). Proc. Human-Computer Interaction. INTERACT '87, Elsevier Science (North-Holland), Amsterdam, 1987, S. 389-349.

Koch, Reiterer & Tjoa, 1991

Manfred Koch, Harald Reiterer, A Min Tjoa. Software-Ergonomie. Gestaltung von EDV-Systemen - Kriterien, Methoden und Werkzeuge. Springer, Wien, New York, 1991.

Laurel, 1986

Brenda Laurel. Interface as Mimesis. In: Donald A. Norman, Stephen W. Draper (Hrsg.) User Centered System Design. New Perspectives on Human-Computer Interaction. Lawrence Erlbaum, Hillsdale, N.J., 1986, S. 67-85.

Maaß, 1983

Susanne Maaß. Why Systems Transparency? In: Thomas R.G. Green, Stephen J. Payne, Gerrit C. van der Veer (Hrsg.). The Psychology of Computer Use. Academic Press, London, 1983, S. 19-28.

Maaß, Rosson & Kellogg, 1987

Susanne Maaß, Mary Beth Rosson, Wendy A. Kellogg. Benutzerfreundlichkeit, Systemkonsistenz und andere schwer definierbare Prinzipien: Interviews mit Systementwicklern. In: Wolfgang Schönpflug, Marion Wittstock (Hrsg.). Software-Ergonomie '87. Nützen Informationssysteme dem Benutzer? Berichte des German Chapter of the ACM, 29, Teubner, Stuttgart, 1987, S. 417-427.

Nielsen, 1989

Jakob Nielsen (Hrsg.). Coordinating User Interfaces for Consistency. Academic Press, Boston, San Diego u.a., 1989.

Nievergelt, 1983

Juerg Nievergelt. Die Gestaltung der Mensch-Maschine-Schnittstelle. In: Ingbert Kupka (Hrsg.). 13. GI-Jahrestagung, Informatik Fachberichte, 73, Springer: Berlin u.a., 1983, S. 41-50.

Oberquelle, 1991

Horst Oberquelle. MCI - Quo vadis? Perspektiven für die Gestaltung und Entwicklung der Mensch-Computer-Interaktion. In: David Ackermann, Eberhard Ulich (Hrsg.). Software-Ergonomie '91. Benutzerorientierte Software-Entwicklung. Berichte des German Chapter of the ACM, 33, Teubner, Stuttgart, 1991, S. 9-24.

Oberquelle, 1994

Horst Oberquelle. Situationsbedingte und benutzerorientierte Anpaßbarkeit von Groupware. In: Anja Hartmann, Thomas Herrmann, Markus Rohde, Volker Wulf (Hrsg.). Menschengerechte Groupware - Software-ergonomische Gestaltung und partizipative Umsetzung. Berichte des German Chapter of the ACM, 42, Teubner, Stuttgart, 1994, S. 31-49.

Oppermann, 1994

Reinhard Oppermann. Individualisierung von Benutzungsschnittstellen. In: Edmund Eberle, Horst Oberquelle, Reinhard Oppermann (Hrsg.). Einführung in die Software-Ergonomie. Gestaltung graphisch-interaktiver Systeme: Prinzipien, Werkzeuge, Lösungen. de Gruyter, Berlin, 1994 (im Druck).

Rutkowski, 1982

Chris Rutkowski. An Introduction to the Human Applications Standard Computer Interface. Part 1: Theory and Principles. Byte, 7, 10, October, 1982, S. 291-310.

Semmer, 1990

Norbert Semmer. Stress und Kontrollverlust. In: Felix Frei, Ivars Udris (Hrsg.). Das Bild der Arbeit, Huber, Bern, S. 190-207.

Shneiderman, 1983

Ben Shneiderman. Direct Manipulation: A Step Beyond Programming Languages. IEEE Computer, 16, 8, 1983, S. 57-69.

Spinas, Waeber & Strohm, 1990

Philipp Spinas, Daniel Waeber, Oliver Strohm. Kriterien benutzerorientierter Dialoggestaltung und partizipative Softwareentwicklung: eine Literaturlaufarbeitung. Projektbericht 1 zum Forschungsprojekt BOSS, Zürich, 1990.

Troy, 1981

Norbert Troy. Zur Bedeutung der Streßkontrolle. Experimentelle Untersuchungen über Arbeit unter Zeitdruck. Dissertation ETH Nr.6795, ADAG, Zürich, 1981

Ulich, 1991

Eberhard Ulich. Arbeitspsychologie. Poeschel, Stuttgart, 1991.

Wandmacher, 1993

Jens Wandmacher. Software-Ergonomie. de Gruyter, Berlin, 1993.

Winograd & Flores, 1986

Terry Winograd, Fernando Flores. Understanding Computers and Cognition. A New Foundation for Design. Ablex, Norwood, N.J., 1986.

Wulf, 1994

Volker Wulf. Das Konzept gestufter Metafunktionen - ein Mittel zur Moderation von Konflikten in Groupware. In: Anja Hartmann, Thomas Herrmann, Markus Rohde, Volker Wulf (Hrsg.). Menschengerechte Groupware - Software-ergonomische Gestaltung und partizipative Umsetzung. Berichte des German Chapter of the ACM, 42, Teubner, Stuttgart, 1994, S. 125-150.

Dieser Artikel wurde als Bericht FBI-HH-B-170/95 des Fachbereichs Informatik, Universität Hamburg, 1995, veröffentlicht.
