

Ergonomie-Handbuch zur Gestaltung virtueller Lerneinheiten

Version 1.0.4 (Arbeitsversion)

Ronald Hartwig, Johannes K. Triebe, Michael Herczeg

Institut für Multimediale und Interaktive Systeme

Medizinische Universität zu Lübeck

Inhaltsverzeichnis

| | | |
|----------|---|-----------|
| 1 | Einführung | 7 |
| 2 | Grundlagen | 9 |
| 2.1 | Aufgaben und Ziele der Ergonomie | 9 |
| 2.2 | Computergestütztes multimediales Lernen | 11 |
| 2.2.1 | Gestaltungsziele | 13 |
| 2.2.2 | Instruktionsdesign als Gestaltung kognitiver Prozesse . | 16 |
| 2.2.3 | Anforderungen aus dem Lernprozess | 18 |
| 2.2.4 | Zusammenfassung | 20 |
| 3 | Methodisches Rahmenkonzept | 23 |
| 3.1 | Prozessmodell | 23 |
| 3.1.1 | Die Prozessphasen im Überblick | 23 |
| | Phase 1: Analyse | 23 |
| | Phase 2: Konzeption | 25 |
| | Phase 3: Design | 25 |
| | Phase 4: Implementierung | 26 |
| | Phase 5: Evaluierung | 26 |
| | Phase 6: Wartung | 27 |
| 3.1.2 | Iterative Entwicklung | 27 |
| 3.1.3 | Mögliche Aufgabenteilung | 27 |
| | Fachautoren | 27 |
| | Pädagogen/Didaktiker | 28 |
| | Designer/Ergonomen | 29 |
| | Multimedia Producer | 29 |
| 3.2 | Der Herstellungsprozess | 29 |
| 3.2.1 | Analyse | 29 |
| | Wo und wie bekomme ich Informationen über den Lern- raum? | 30 |
| | Was sind wichtige Merkmale? | 31 |
| | Wie werden Ergebnisse der Analyse dokumentiert? . . . | 33 |
| 3.2.2 | Konzeption | 33 |
| | Lernziele bestimmen | 34 |
| | Ableitung besonderer Anforderungen aus dem Nut- zungskontext | 34 |
| | Auswahl des didaktischen Konzeptes | 34 |
| | Entwurf eines Drehbuches | 34 |
| 3.2.3 | Design | 35 |

| | | |
|----------|---|-----------|
| | Prototyping | 35 |
| 3.2.4 | Implementierung | 35 |
| | Funktionale Tests | 36 |
| 3.2.5 | Evaluierung | 36 |
| | Effektivität | 36 |
| | Effizienz | 37 |
| | Zufriedenheit der Benutzer | 37 |
| | Evaluationsansätze | 39 |
| 3.2.6 | Wartung von Lehreinheiten | 41 |
| 4 | Detaildarstellung grundlegender Regeln | 43 |
| 4.1 | Regeln zur Dialoggestaltung | 43 |
| 4.1.1 | Aufgabenangemessenheit | 43 |
| | Definition | 43 |
| | Beispiel Aufgabenangemessener Dialog | 44 |
| | Konsequenzen | 44 |
| 4.1.2 | Selbstbeschreibungsfähigkeit | 45 |
| | Definition | 45 |
| | Beispiel Unangemessene Ausdrucksweise | 46 |
| 4.1.3 | Steuerbarkeit | 46 |
| | Definition | 46 |
| | Beispiel Gute Steuerbarkeit | 47 |
| 4.1.4 | Fehlertoleranz | 47 |
| | Definition | 47 |
| | Beispiel Mangelhafte Fehlermeldung | 48 |
| 4.1.5 | Erwartungskonformität | 48 |
| | Definition | 48 |
| | Beispiel Erwartungskonformer Dateiübertragungsdialog | 49 |
| 4.1.6 | Individualisierbarkeit | 50 |
| | Definition | 50 |
| | Beispiel Gute Individualisierbarkeit bei Netscape | 50 |
| 4.1.7 | Lernförderlichkeit | 50 |
| | Definition | 50 |
| | Beispiel Lernförderlichkeit durch grafische Darstellung | 52 |
| 4.2 | Regeln zur Informationsdarstellung | 53 |
| 4.2.1 | Darstellungsprinzipien und Grundregeln | 53 |
| | Beispiel | 54 |
| 4.2.2 | Klarheit | 56 |
| | Definition | 56 |
| | Beispiel Unklare Strukturierung | 56 |
| 4.2.3 | Unterscheidbarkeit | 58 |
| | Definition | 58 |
| | Beispiel Unterscheidung von Links | 58 |
| 4.2.4 | Kürze | 58 |
| | Definition | 58 |
| | Beispiel Überladene Web-Seite | 58 |
| 4.2.5 | Konsistenz | 60 |

| | |
|--|------------|
| Definition | 60 |
| 4.2.6 Entdeckbarkeit | 60 |
| Definition | 60 |
| 4.2.7 Lesbarkeit | 61 |
| Beispiel Großbuchstaben | 61 |
| 4.2.8 Verständlichkeit | 63 |
| Definition | 63 |
| Beispiel Unverständliche Icons | 63 |
| 5 Entwurfscheckliste | 65 |
| 5.1 Schrift | 65 |
| 5.2 Ausdruck | 66 |
| 5.3 Gestaltung | 67 |
| 5.4 Benutzerunterstützung | 69 |
| 5.5 Information des Benutzers | 70 |
| 6 Hintergrund: Bildschirmarbeit | 73 |
| 6.1 Die Bildschirmarbeitsverordnung | 73 |
| 6.2 Systemische Betrachtungen zur Bildschirmarbeit | 75 |
| 6.2.1 Ein-/Ausgabe-Schnittstelle | 80 |
| 6.2.2 Dialog-Schnittstelle | 81 |
| 6.2.3 Werkzeug-Schnittstelle | 81 |
| 6.2.4 Organisationsschnittstelle | 83 |
| 6.2.5 Ein Vorschlag zur Ergänzung des IFIP-Modells | 83 |
| 6.3 Software-ergonomische Gestaltungsgrundsätze und Normen | 85 |
| 6.4 Referenzkonzept ISO 9241 | 86 |
| 6.4.1 Anforderungen an die Gebrauchstauglichkeit - Leitsätze (ISO 9241-11) | 91 |
| 6.4.2 Ein-/Ausgabeschnittstelle: Informationsdarstellung (ISO 9241-12) | 93 |
| 6.4.3 Dialogschnittstelle: Grundsätze der Dialoggestaltung (ISO 9241-10) | 94 |
| 6.4.4 Benutzerführung (ISO 9241-13) | 96 |
| 6.4.5 Dialogführung mittels Menüs (ISO 9241-14) | 97 |
| 6.4.6 Dialogführung mittels Kommandosprachen (ISO 9241-15) | 97 |
| 6.4.7 Dialogführung mittels direkter Manipulation (ISO 9241-16) | 98 |
| 6.4.8 Dialogführung mittels Bildschirmformularen (ISO 9241-17) | 99 |
| 6.5 Sonstige Richtlinien, Guidelines und Styleguides | 100 |
| 6.6 Überprüfung von Software | 103 |
| 6.7 Prüfverfahren - eine Kurzübersicht | 110 |
| Literaturverzeichnis | 113 |
| Abbildungsverzeichnis | 121 |

1 Einführung

Zur Gestaltung und Realisierung von multimedialen und interaktiven Lehreinheiten für ein virtuelles Studium ist es notwendig, grundlegende ergonomische und andere gestalterische Randbedingungen in Form eines Handbuchs darzustellen.

Auf diese Weise soll ein gestalterisches Regelwerks zur Sicherung ergonomischer Qualitäts- und Funktionsanforderungen entwickelt werden. Wichtige ergonomische Aspekte sind:

- hardware-ergonomische Randbedingungen (Arbeitsplatzgestaltung)
- Informationscodierung (z.B. Schriften, Farbe, Graphik, Anordnung)
- Interaktionsformen (z.B. formale Sprachen, Menüs, Funktionstasten, Formulare)
- Dialogparadigmen (z.B. Menü-/Maskensysteme, direkte Manipulation, Hypermedia)
- Unterstützungssysteme (z.B. Hilfesysteme, Historysysteme, Individualisierung)
- Zeitverhalten (z.B. Ausgabezeiten, Antwortzeiten, Animationsverhalten)

Das Ergonomie-Handbuch berücksichtigt jeweils aktuell gültige Empfehlungen, Normen und Verordnungen im Zusammenhang mit Ergonomie und Arbeitsplatzvorschriften und formuliert software-ergonomische Mindestanforderungen an Lehr- und Lernarbeitsplätze.

Ein solches Regelwerk soll die Autoren von Studienmodulen bei der Entwicklung von neuen Lehr- und Lernformen sowie den darauf aufbauenden Lehreinheiten unterstützen. Abweichungen in explizit begründeten Einzelfällen von definierten Regeln sollen möglich sein. Die gestalterischen Freiheiten bei der Erstellung von Lehreinheiten sollen, soweit sinnvoll und dem Lernen zuträglich, erhalten werden.

Das Handbuch kann als eine Grundlage für Qualitätssicherungsmaßnahmen für selbst erstellte oder importierte Lehreinheiten dienen. Auf diese Weise wird eine Mindestqualität von Lehreinheiten abgesichert. Mit der hier vorgelegten ersten Arbeitsversion soll für die Ersteller von Lernmodulen und Lerneinheiten ein Orientierungsrahmen vermittelt werden, dessen Berücksichtigung das Auftreten schwerwiegender ergonomischer Mängel verhindern kann oder zumindest unwahrscheinlich macht.

Eine Warnung erscheint allerdings angebracht: Ergonomie war noch nie eine Sammlung einfacher „Patentrezepte“, deren strikte Befolgung ein benutzerfreundliches Produkt garantiert. Nach unserem Verständnis sollte sie dies auch nie sein wollen (vgl. [Herczeg 94] S.VIII). Jedem Entwickler stellt sich deshalb bei der Gestaltung seines Lehrangebotes nach wie vor die Aufgabe, die Hinweise und Empfehlungen dieses Handbuches an die Besonderheiten des von ihm angebotenen Stoffes und seiner Zielgruppe anzupassen.

Nicht zuletzt ist an dieser Stelle daran zu erinnern, daß den ergonomischen Aspekten eines virtuellen Studienangebots in der Regel die pädagogisch-didaktischen Fragen vorangeordnet sind, und daß andererseits auch die aus der schließlich gewählten technischen Plattform sich ergebenden Rahmenbedingungen zu berücksichtigen sind. Deshalb ist es wichtig, der Interaktion dieser drei Gestaltungsfaktoren Rechnung zu tragen und auf längere Sicht ein generelles Vorgehensmodell für die Entwicklung und Gestaltung von Lehr-/Lerneinheiten zu erarbeiten.

2 Grundlagen

2.1 Aufgaben und Ziele der Ergonomie

Die Ergonomie befaßt sich mit der Anpassung von (Arbeits-)Systemen, Werkzeugen und Maschinen (Gerätschaften) sowie Produkten (Gegenständen) an die Besonderheiten der mit ihnen in Kontakt tretenden, insbesondere arbeitenden Menschen (siehe als Konkretisierung auch Abb. 2.1). Trotz der enormen, evolutionär und historisch deutlich gewordenen Anpassungs- und Lernfähigkeit des Menschen besteht das vorrangige Ziel der Ergonomie somit darin, diese Fähigkeiten produktiv wirksam werden zu lassen und nicht zu vergeuden („Persönlichkeitsförderung“). Um dieses herum gruppieren sich weitere Ziele, die sich unter Gesichtspunkten wie Zuverlässigkeit, Sicherheit und Gesundheit, sowie Leistungsoptimierung und Effizienzverbesserung zusammenfassen lassen.

Die zu berücksichtigenden Besonderheiten des Menschen betreffen sowohl physische Ausstattung und Leistungsvermögen (wie z.B. Skelett und Muskulatur oder sinnesphysiologische Kennwerte), als auch seine gesamte psychische Ausstattung, insbesondere Prozesse der Wahrnehmung und Aufmerksamkeit, des Behaltens und Wiedererinnerns, der Begriffsbildung und Kommunikation mit Hilfe symbolischer (auch nicht-sprachlicher) Repräsentationen, des Denkens, Handelns und Problemlösens. Nicht zuletzt gehören hierher aber auch Ansprüche und Wertvorstellungen, die gesellschaftshistorischen Veränderungen unterliegen, aber doch in erheblichem Maße das Bild des Menschen von sich selbst - und damit z.B. von dem, was „angemessene“ Anforderungen an ihn sind - bestimmen.

Mit dem Computer hat der Mensch nunmehr ein einzigartiges neues Werkzeug bzw. eine universale, mittels Software an ein breites Spektrum von Aufgaben anpaßbare Maschine entwickelt, die zunehmend sein Denken und Handeln innerhalb und außerhalb des Kontextes von Arbeit beeinflußt oder sogar dominiert.

So ergab sich auch für die Ergonomie ein neues Arbeitsfeld von ständig wachsender Bedeutung. Nicht nur *hardware*-ergonomische Fragen, die im wesentlichen die körperlichen Folgen des Computereinsatzes betreffen, sind zu beantworten. Die meisten und komplexesten Probleme stellen sich vielmehr der sog. Software-Ergonomie, die sich mit der menschengerechten Gestaltung computer-unterstützter Arbeitssysteme befaßt und dabei der Tatsache Rechnung tragen muß, daß deren Eigenschaften grundlegend und in einem außerordentlichen Maße von den ergonomischen Qualitäten der eingesetzten Software determiniert werden.

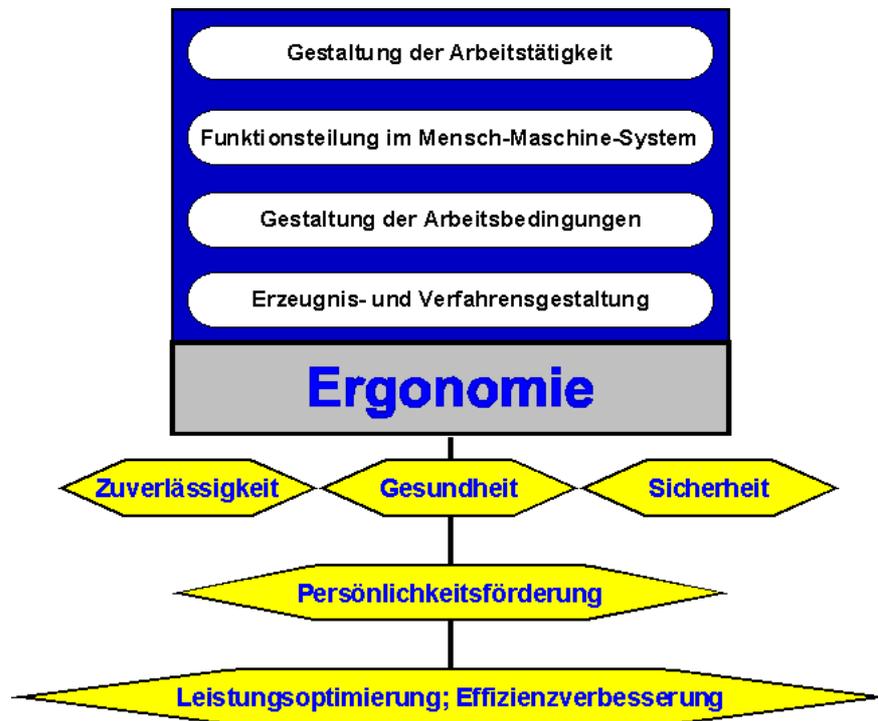


Abbildung 2.1: Aufgabenschwerpunkte und Ziele der Ergonomie

Software-Ergonomie

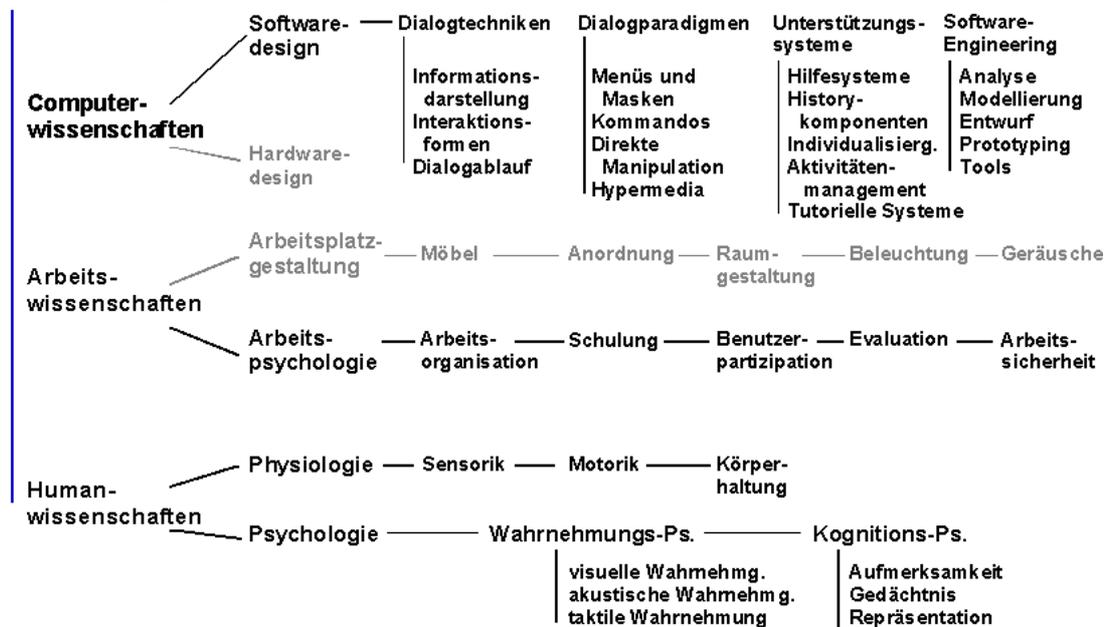


Abbildung 2.2: Software-Ergonomie als multi-disziplinäres Arbeitsgebiet (nach [Herczeg 94])

Die Software-Ergonomie hat sich — vor allem während der vergangenen zwanzig Jahre — rasch zu einem multi-disziplinären Fach mit einer kaum noch überschaubaren Fülle z.T. hoch spezialisierter Fragestellungen entwickelt (siehe Abb. 2.2). Ein kurzer Abriß der historischen Entwicklung und eine Darstellung der mittlerweile ausdifferenzierten Wissenschaftslandschaft findet sich bei Streitz (siehe [Streitz 88]).

Entsprechend vielfältig werden auch die Themen und Ratschläge für die ergonomische Gestaltung des Lehrangebotes der VFH sein, die im Laufe der Zeit in dieses Ergonomie-Handbuch — als work in progress — aufgenommen und in Seminaren und Beratungsgesprächen vermittelt werden müssen.

2.2 Ergonomische Anforderungen des computergestützten multimedialen Lernens

Im folgenden sei versucht, einige grundlegende Gesichtspunkte herauszuarbeiten, die für die ergonomische Gestaltung von computergestützten multimedialen Lehr- und Lernsystemen (CMMLLS) von zentraler Bedeutung sind. Aus ihnen ergeben sich eine Reihe von Gestaltungsprinzipien, die sich unter Umständen zwar in unterschiedlicher Weise realisieren lassen, aber doch mit großer Wahrscheinlichkeit zur Entstehung einer Lernumgebung (und Benutzungsoberfläche) beitragen, die die Lernenden bei ihrer Aufgabe unterstützt.

Diese Unterstützung der Lernenden hat dabei im Vordergrund aller Gestaltungsansätze zu stehen. Selbstverständlich spielt auch die Unterstützung der Lehrenden - wie auch aller der Studienorganisation dienenden Verwaltungsaufgaben - eine wichtige Rolle für künftig bereitzustellende CMMLLS; sie bleibt aber in diesem Handbuch vorerst außer Betracht, kann allerdings Gegenstand des seitens des IMIS bereitgestellten Beratungsangebots sein.

Letzten Endes geht es für die VFH um die Entwicklung und (softwaretechnische) Realisierung eines kompletten, in sich konsistenten Systems, das heute vielfach metaphorisch als „Lernraum“ bezeichnet wird. Räumliche Metaphern haben sich für das Erinnern von und Nachdenken über komplexe Zusammenhänge verschiedener Art als nützlich erwiesen (vgl. [Dutke 94], S.113ff.). Trotz der Bemühungen und vielversprechenden Ankündigungen zahlreicher Software-Produzenten stehen geeignete Systeme bisher nicht zur Verfügung, was angesichts der in einer Studie von Nagl et al. ([Nagl et al. 99]) dargestellten Probleme wenig erstaunt. Mit der für die VFH sich mittlerweile abzeichnenden Entscheidung für „Blackboard“ als einem der auf dem Markt gegenwärtig angebotenen Systeme entsteht immerhin Klarheit darüber, innerhalb welcher gestalterischen und technischen Rahmenbedingungen das zu entwickelnde Lehr-Lern-Angebot zu realisieren sein wird. Ein „Lernraum“ i.e.S. entwickelt sich aber daraus erst im Zuge der Ausgestaltung von Lernmodulen und -einheiten (einschließlich der zu ihrer Benutzung erforderlichen Hilfsmittel) nach einheitlichen und - wo immer möglich - verschiedenen Lerninhalten übergeordneten Gesichtspunkten.

Die häufig zu beobachtende Gleichsetzung des „Lernraumes“ mit einem bestimmten (käuflich zu erwerbenden) Softwaresystem verstellt auch leicht den

Blick auf die bei seiner Entwicklung zu berücksichtigenden psychologischen Probleme.

Der „Lernraum“ läßt sich prinzipiell auffassen als

- (a) individuelle mentale Repräsentation - oder besser: mentales Modell - aller dem Lernenden zur Verfügung stehenden Handlungsmöglichkeiten (Vorgehensweisen) und -ergebnisse;
- (b) multi-mediale Präsentation als Veranschaulichung dieser Handlungsmöglichkeiten und -ergebnisse (z.B. mit Hilfe von Servern und via Intra- und/oder Internet).

Die Aufgabe der Ergonomie besteht damit in einer Optimierung der Beziehung zwischen diesen beiden „Abbildungen“, indem (b) aufgrund der Erkenntnisse über die kognitiven Funktionen des Menschen so gestaltet wird, daß die Schwierigkeiten der individuellen Übersetzung nach (a) minimiert werden. Hierbei sind vielfach sogenannte „konzeptuelle Modelle“ von Nutzen, die gezielt (von Softwareentwicklern, Ergonomen, Designern) zur Veranschaulichung der zwischen verschiedenen Systemkomponenten bestehenden Zusammenhänge konzipiert werden. Beispiele hierfür bilden etwa die Ausgestaltung der Desktop-Metapher in den verschiedensten, im wesentlichen auf „direkter Manipulation“ basierenden Softwaresystemen, oder - als Detaillösung - die „Radio-Button“-Analogie zur Auswahl zwischen einander ausschließenden Alternativen im Windows-Styleguide ([Microsoft 95]).

Für den Bereich der CMLLS haben sich noch keine allgemein anerkannten, spezifischen Metaphern herausgebildet, weshalb hier nur auf die Bedeutung dieses Gesichtspunktes hingewiesen werden kann. Die Funktionen mentaler Modelle für die Entstehung menschlichen Wissens und Verstehens, sowie in ihrer Anwendung auf software-ergonomische Probleme, sind bei Dutke ([Dutke 94]1994) ausführlich dargestellt. Mentale Modelle sind

- subjektive (Re-)Konstruktionen von Sachverhalten oder Situationen,
- basieren oft auf Analogien (Metaphern, Heuristiken), mit deren Hilfe Vorwissen auf neue Probleme übertragen wird,
- und werden dynamisch für bestimmte Zwecke gebildet, die sich zusammenfassend als Bewältigung der Anforderungen einer aktuellen Situation kennzeichnen lassen.

Oft nicht explizierte (konzeptuelle), sondern allenfalls aus dem Resultat erschließbare mentale Modelle von den Lernenden, dem Lernprozeß und Lernerfolg sind häufig auch das, was den Gestaltungsansätzen der Entwickler (Autoren und Designer) von Lerneinheiten zugrunde liegt. Unvollständige und nicht beschriebene Modelle verführen leicht zu Detaillösungen, die je nach dem vermittelnden Stoff, der Verfügbarkeit (multi-)medialer Darstellungsmöglichkeiten sowie dem eigenen Geschick im Umgang mit bestimmten Gestaltungstools gewählt werden, aber die innere Konsistenz und Durchgängigkeit eines systematisch angelegten Gestaltungskonzepts vermissen lassen.

Deshalb ist es notwendig, eine Reihe grundsätzlicher Überlegungen und Entscheidungen anzusprechen, die den Ausgangspunkt jedes Vorgehens bilden müssen, das den Ansprüchen des weithin geforderten „instructional design“ entspricht.

Ballstaedt ([Ballstaedt 97], S.12) definiert **didaktisches Design** als „die planmäßige und lernwirksame Entwicklung von Lernumgebungen (von der Bedarfsanalyse bis zur Evaluation) auf wissenschaftlicher Grundlage.“ Psychologie, Pädagogik und Ergonomie sind - in ihrem interdisziplinären Zusammenwirken - die dabei in erster Linie angesprochenen Wissenschaften.

2.2.1 Gestaltungsziele

Übergeordnete Gestaltungsziele und -anforderungen ergeben sich aus der Entscheidung für eine bestimmte Lehr-Lern-Konzeption, die gewissermaßen die didaktische Gesamtzielsetzung repräsentiert; sie bildet den Rahmen, der danach zielgruppenspezifisch und dem Lernstoff angepaßt konkretisiert werden kann und muß.

Im Vordergrund der Diskussion um Formen der Wissenserarbeitung und -vermittlung, die gerade unter der Perspektive des „life-long learning“ von besonderer Bedeutung sind, stehen heute zunehmend (oder sogar ausschließlich) sogenannte „konstruktivistische“ Positionen (vgl. [Gerstenmaier und Mandel 95]). Sie lassen sich zusammenfassend wie folgt charakterisieren: „Die konstruktivistische Lehr-Lernphilosophie stellt den Lernenden in den Mittelpunkt. Sie definiert Lernen als einen aktiv-konstruktiven Prozeß, der individuell und situationsspezifisch geprägt ist, was die Lernziele, Inhalte und den Lernverlauf betrifft. Der Lehrende wird als Teil einer kreativen Lernumwelt gesehen (traditionelle einschließlich technologische Medien), die dem Lernenden selbstgesteuertes, problemorientiertes und kooperatives Lernen ermöglichen soll“ ([Issing 99], S.2). Kommunikation und Kooperation - insbesondere auch der Lernenden untereinander - stehen im Mittelpunkt dieses Prozesses, bei dem Wissen fortlaufend gemeinschaftlich im Hinblick auf die aktuelle Anforderungssituation (re-)konstruiert wird, wobei das individuelle Lernen durch die dabei unverzichtbare Übernahme von Sichtweisen und Rollen anderer Personen besonders gefördert wird (perspective taking / role taking).

Die traditionelle Lehr-Lernphilosophie ist demgegenüber nach Issing (a.a.O.) „gekennzeichnet durch die Vorrangstellung des Lehrens als einem Wissenstransport vom Lehrenden zum Lernenden. Der Lernende wird im wesentlichen als rezeptives System verstanden, das das neue Wissen abspeichert und in der Lage sein soll, dieses Wissen abrufbar zu reproduzieren und auf neue Zusammenhänge zu übertragen.“ Dabei erscheint uns Issings Hinweis erwähnenswert, daß diese Position „in der Vergangenheit in vielen Bereichen effektiv und ausreichend“ gewesen sei und „auch heute noch in vielen Sektoren bei der Vermittlung von Grundlagenwissen und beim Erwerb bestimmter Grundfertigkeiten ihre Berechtigung“ habe ([Issing 99], S.2). Inwieweit der Erwerb entsprechender Grundfertigkeiten auch im Bereich des von der VFH zur

Verfügung zu stellenden Lernangebots eine Rolle spielt, sollte daher von Fall zu Fall überprüft werden.

Einige wichtige Grundsätze der konstruktivistischen Lehr-Lernphilosophie sollen hier stichwortartig die Gestaltungsziele verdeutlichen:

Experiential Learning

- Aufgaben und Inhalte orientieren sich an ihrer Bedeutung für die Lernenden, ihre Wünsche und Bedürfnisse, daher sollten möglichst verschiedene Angebote zur Auswahl gestellt werden
- Persönliches Engagement, Eigeninitiative, Selbstkontrolle bis hin zur Selbst- anstelle Fremdbeurteilung sollen gefördert werden
- Die Lernumgebung ist als „low-risk environment“ zu organisieren: experimentierfreundlich, fehlertolerant, Sackgassen und nicht mehr rückgängig zu machende Entscheidungen sind zu vermeiden

Situated Learning

- Lernen vollzieht sich eingebettet in die Aktivität der Lernenden, den Kontext und die kulturellen Rahmenbedingungen
- soziale Interaktion und Kooperation sind zentraler Bestandteil der Lernprozesse
- der Kontext sollte so „authentisch“ wie möglich sein
- aus der Kooperation mit Experten soll sich eigenes Expertentum entwickeln

Anchored Instruction

- Verankerung des Lernens (Ausgangspunkt) in einem konkreten Fallbeispiel bzw. einer realistischen Problemdarstellung
- Vermittlung dieses Ankerpunktes bevorzugt mittels Video oder ähnlich anschaulicher Darstellungstechniken
- Das Lernmaterial muß exploratives Arbeiten unterstützen
- Die Lernaktivität zielt auf Entwicklung von Wegen und Verfahren (Tools) zur Lösung komplexer, realitätsnaher Probleme

Im Einsatz vernetzter, computergestützter, interaktiver und multimedialer Lehr-Lern-Systeme liegen u.E. prinzipiell große Chancen, konstruktivistische Ansätze in effizienter Weise zu unterstützen. Dies setzt allerdings voraus, daß es gelingt

- **Multimedialität** gezielt im Sinne der Vermittlung unterschiedlicher Sichtweisen - aber auch unterschiedlicher Formen sinnlicher Erfahrung - einzusetzen;

- **Interaktivität** zur aktiven Erforschung komplexerer Zusammenhänge und zur darauf aufbauenden Modellbildung zu nutzen, wobei die Simulationsfähigkeit des Computers zur Veranschaulichung der Einflüsse einzelner Modellparameter und ihres dynamischen Zusammenwirkens verwendet werden kann, und eine rasche Rückmeldung über die Konsequenzen unterschiedlicher Modellannahmen (d.h. Interpretationen der Realität) ermöglicht.
- **Vernetzung** schließlich muß sowohl die gezielte Recherche zusätzlicher, bisher nicht berücksichtigter Sachverhalte und Sichtweisen (und die kritische Wertung unterschiedlicher Quellen) herausfordern, als auch verschiedenartige Kommunikations- und Kooperationsformen (wie Chat, Whiteboard, Application Sharing / Joint Working, gegenseitige Hilfestellung, Coaching / Tutoring, Rat und Hilfe von Experten) wirkungsvoll unterstützen.

Der vorwiegend „virtuelle“ Charakter dieses immer noch neuartigen - aber bereits weltweit im Versuchsstadium befindlichen - Lernens und Studierens darf jedenfalls nicht dazu führen, daß die Beteiligten gerade bezüglich der so wichtigen *sozialen* Vermittlungskomponente des Wissenserwerbs benachteiligt werden.

Ein wichtiger *prozeduraler* Aspekt ist u.E. ergänzend als unverzichtbarer Bestandteil konstruktivistischer Ansätze hervorzuheben: Die Kontrolle über den Lernprozeß muß weitgehend den Lernenden selbst überantwortet werden. Bei Doherty ([Doherty 98]) findet sich eine knappe Literaturübersicht, die sowohl die historische Entwicklung dieses Prinzips der „learner-controlled instruction“ (LCI) als auch seine aktuelle Anwendung auf Studienangebote im Internet umfaßt.

Die Kontrolle über den Lernprozeß bezieht sich nach Doherty ([Doherty 98], S.2) auf die folgenden Gesichtspunkte:

- „In its broadest sense, learner control is the degree to which a learner can direct his/her own learning experience.
- More specifically, learner control is the degree to which individuals control the path, pace, and/or contingencies of instruction.
- Yet, the meaning of learner control has evolved over time to include the characteristics of new learning paradigms as well as new technologies. Interactive instructional systems, such as learning networks, make it possible to provide learners with control over
 - depth of study,
 - range of content,
 - number and type of delivery media,
 - and time spent on learning.

With these options, learners can tailor the learning experience to meet their specific needs and interests“ (Layout gegenüber Doherty verändert). LCI kann

so erheblich zum Aufbau von intrinsischer Motivation und Zufriedenheit der Lernenden beitragen.

Doherty betont (unter Verweis auf Frank Wydra als einen der Pioniere dieser Auffassung), daß LCI die Lehrenden keineswegs ihrer Verantwortung enthebt, sondern sich der Schwerpunkt des Gestaltungsprozesses von der Instruktion i.e.S. auf das sorgfältige Design einer entsprechenden Lernumgebung verlagert. „...it is necessary to emphasize the point that LCI is not learner anarchy. There *is* control. But it is on the environmental level. There *is* direction. But it is a function of the design of the environment. There *is* learner freedom. But it is within the consequences and resources that have been built into the environment“ ([Wydra 80], zit. nach [Doherty 98], S.2).

Konstruktivistische Ansätze und ein differenziertes Angebot zur technischen Unterstützung der Kontrolle ihres Lernprozesses durch die Lernenden selbst sind nach unserer Auffassung für die künftige Zielgruppe des virtuellen Studienangebots der VFH besonders geeignet: Das berufsbegleitende, auf fortlaufende Wissensaktualisierung und die Bewältigung neuartiger Anforderungen gerichtete Lernen erwachsener Menschen ist durch ein computergestütztes, nach didaktischen und ergonomischen Prinzipien gestaltetes (weitgehend asynchrones) „Lernen im Netzwerk“ ([Harasim 95]) wirksam zu unterstützen.

2.2.2 Instruktionsdesign als Gestaltung kognitiver Prozesse

Alle Formen menschlichen Lernens entwickeln sich aus dem Zusammenspiel grundlegender psychischer (insbesondere kognitiver) Prozesse zur Steuerung des in der Vorstellung (Problemlösen) oder tatsächlich vollzogenen Handelns im sozialen Kontext, wie dies in Abbildung 2.3 veranschaulicht wird.

Die hier gegebene Übersicht soll in Erinnerung rufen, daß Instruktionsdesign und ergonomische Gestaltungsansätze stets auch als Maßnahmen zur Unterstützung und Beeinflussung der beteiligten psychischen Prozesse zu verstehen sind und von dort her ihren Begründungszusammenhang beziehen. Für die Entwickler von Lerneinheiten empfiehlt es sich daher stets, eine Zusammenstellung und Systematisierung der selbst gewählten Gestaltungsmittel anhand dieser psychologischen Kategorien zu versuchen. Leitfragen wären dabei etwa:

Mit welchen - vorwiegend optischen und akustischen - Gestaltungsmitteln

- gliedere ich das Wahrnehmungsfeld (z.B. Abstände, Zentrierungen, Ähnlichkeiten);
- steuere ich die Aufmerksamkeit (z.B. Farben, Helligkeitsunterschiede und andere Auszeichnungen, Bewegungen);
- unterstütze ich die Entstehung gedanklicher Ordnungsstrukturen, kognitiver Landkarten u.ä. (z.B. Mindmaps, Begriffshierarchien, Ablaufdiagramme);
- läßt sich das längerfristige Erinnern erleichtern (z.B. Ähnlichkeiten von Lern- und Prüfsituation);

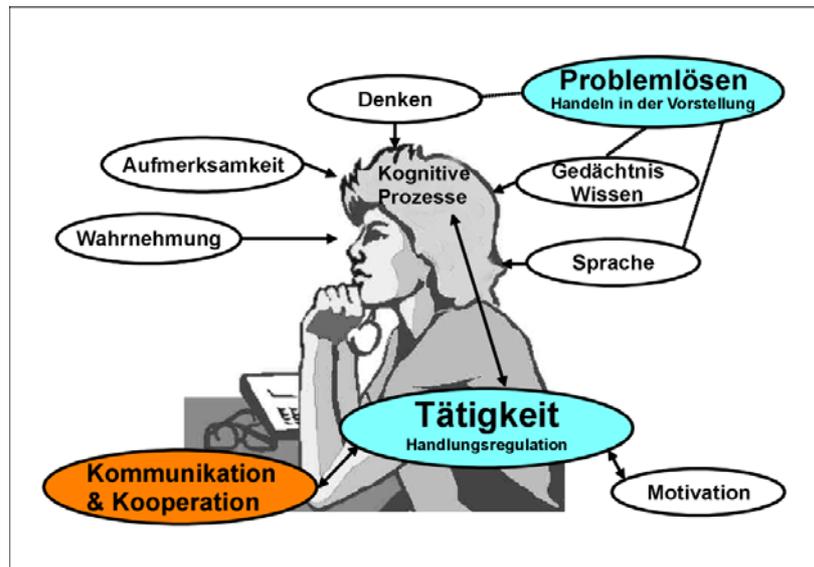


Abbildung 2.3: Lernen im Zusammenwirken psychischer Prozesse

- ... etc.?

Das vorliegende Handbuch Ergonomie kann und will kein Übersichtsreferat der Kognitiven Psychologie sein. Entwickler von Lerneinheiten sollten aber bei Bedarf auf eine sehr differenzierte Darstellung zurückgreifen, die Online zur Verfügung steht ([Klein 99]) und in weiten Teilen aus der Aufbereitung eines der anerkanntesten Lehrbücher zur Kognitiven Psychologie ([Wessels 84]) besteht.

Überdies hat Ballstaedt ([Ballstaedt 97]) zur Gestaltung von Lernmaterialien, d.h. der (multi-)medialen Aufbereitung des Lernstoffs, eine sehr gründliche und umfassende Darstellung geschrieben, die wir allen Entwicklern von Lerneinheiten als ausgesprochen praxisnahe Referenz empfehlen können. Er behandelt dort im Hinblick auf ein „didaktisches Design“ die wichtigsten Darstellungsformen:

- Texte (einschließlich gesprochener Sprache)
- Charts
- Tabellen
- Diagramme
- Abbilder (einschließlich Film/Video)
- Piktogramme

in ihrer ganzen Variationsbreite.

Dabei erläutert er auch die darstellungsspezifischen, dem Lernprozeß zugrundeliegenden kognitiven Verarbeitungsformen und gibt nicht zuletzt konkrete Gestaltungshinweise, die als praktische Umsetzung ergonomischen Wissens gelten können.

2.2.3 Anforderungen aus dem Lernprozess

Neben der Entscheidung für eine bestimmte Lehr-Lern-Konzeption sind übergeordnete Gestaltungsziele auch aus einer näheren Betrachtung einiger verallgemeinerbarer Merkmale von Lernprozessen abzuleiten; sie ergeben sich unabhängig von den Besonderheiten des Lernstoffs bzw. der Domäne des jeweils zu vermittelnden Wissens.

Hier sind vor allem die verschiedenen „Phasen“ oder „Stationen“ zu nennen, die jeder Lernprozess durchläuft, nämlich

- (Vor-)Orientierung;
- Aneignung des (neuen) Wissens;
- Anwendung und
- Übertragung (Transfer);

sowie in der Verallgemeinerung des bezüglich des Lernens selbst neu hinzugewonnenen prozeduralen Wissens

- als Meta-Aspekt: Learning how to learn.

Gute Lehr-Lern-Systeme bieten jeweils spezifische Unterstützung für jede einzelne dieser Phasen und ermöglichen so vollständige Lernhandlungen.

Die Phase der **(Vor-)Orientierung** kann in ihrer Bedeutung für den weiteren Lernprozess nicht hoch genug eingeschätzt werden. In ihr entscheiden die Studierenden, worauf sie ihre Aufmerksamkeit richten wollen, wie sie sich die zum Lernen zur Verfügung stehende Zeit einteilen und wie sie ihre kognitiven Ressourcen einsetzen werden. Sie vergewissern sich ihrer persönlichen Vorkenntnisse und Defizite, entwickeln eigene und/oder übernehmen vorgegebene Zielsetzungen und kommen zu einer Einschätzung des erforderlichen Aufwands und ob dieser sich „lohnt“.

In der Regel ist diese Phase auch ausschlaggebend für den Aufbau einer „intrinsic“ (d.h. der Aufgabe selbst und den eigenen Zielen entstammenden) Motivation, aus der sich im weiteren Verlauf das nötige Durchhaltevermögen speist. Um diese Phase optimal zu unterstützen benötigen die Lernenden daher bereits vor dem Einstieg in eine Lerneinheit Angaben über das Thema, den Umfang, den ungefähren Zeitbedarf, die Vermittlungsform und die notwendigen Eingangsvoraussetzungen, darüber hinaus aber auch Möglichkeiten, sich durch freies Explorieren (im Sinne eines stichprobenartigen Ausprobierens) eigene Vorstellungen von den kommenden Anforderungen zu bilden.

In der Phase der **Aneignung** geht es für die Lernenden darum, für das angebotene neue Wissen Verknüpfungen mit den bereits vorhandenen Kenntnissen zu finden, also letzten Endes neue kognitive Strukturen und Schemata aufzubauen. Dies kann nur durch eine aktive, eigenständige Reorganisation des angebotenen Lernstoffes gelingen und vollzieht sich meist nicht allein durch vorstellungsmäßiges (gedankliches) Operieren. Aus der Arbeit mit traditionellen (Druck-)Medien ist bekannt, daß Lernende hierbei oft mit Unterstreichungen, Markierungen, Randnotizen oder auch Exzerpten und selbstformulierten Zusammenfassungen arbeiten müssen. Auch genügt meist nicht das einmalige Durcharbeiten des Stoffes, vielmehr müssen wichtige Passagen oft mehrfach wiederholt und neu überdacht werden. Es liegt auf der Hand, daß auch im „virtuellen“ Lernraum hierfür geeignete Navigations- und Arbeitsmöglichkeiten (wie z.B. ein sog. Annotationstool) zur Verfügung stehen müssen.

Unsere Beobachtungen von Studierenden bei der Bearbeitung frühzeitig erstellter sogenannter Pilotmodule zeigten z.B., daß eine häufige Arbeitsweise darin bestand, sich angebotene Inhalte ausdrucken zu lassen und diese dann begleitend zur Arbeit am Bildschirm mit Unterstreichungen und Anmerkungen zu versehen.

Lernen beschränkt sich bekanntlich nicht nur auf den Aufbau neuer Wissensstrukturen. Erst durch die **Anwendung** des Gelernten und die damit verbundene Erprobung und Bestätigung seines Nutzens bei der Bewältigung wichtiger und/oder typischer (authentischer) Fragestellungen und Probleme wird es zum stabilen, jederzeit verfügbaren Gedächtnisbesitz. Daher sind geeignete Übungsaufgaben seit jeher ein unverzichtbarer Bestandteil jeder gut gestalteten Lernumgebung; sie dienen der (Selbst-)Überprüfung und dem Nachweis sowohl der Anwendungssicherheit als auch -schnelligkeit des Gelernten in einem möglichst repräsentativen Spektrum von Anwendungssituationen. Daher müssen CMMLLS ohne eine hinreichende Menge abgestufter und variantenreicher Übungsaufgaben als unvollständig angesehen werden.

In der Phase der **Übertragung** (des Transfers) benötigen die Lernenden Hilfen bei der Entdeckung von strukturellen Ähnlichkeiten zwischen für sie ansonsten disparaten Anwendungsgebieten ihres neuerworbenen Wissens, um über die bloße Anwendung des Gelernten hinaus dieses auch auf weitere, eher unerwartete oder ungewöhnliche Bereiche im Sinne eines heuristischen Vorgehens übertragen zu können. Könnerschaft und Expertentum sind weniger im Bereich von Routineaufgaben, sondern besonders bei der Lösung neu auftauchender Probleme eine gefragte Qualifikation. Darüber hinaus stellt sich die Frage des Transfers des Gelernten besonders auch in der Übertragung auf neue soziale Kontexte und Kooperationspartner. Hierfür werden in CMMLLS vielfach komplexere Fallbeispiele, Problemdarstellungen und Planspiele zur gemeinsamen (asynchronen oder synchronen) Bearbeitung benötigt werden, bei denen auch die Transfereffekte unter Umständen bereits einige Zeit zurückliegender Lernperioden erprobt werden können.

Mit dem Meta-Aspekt des „**Learning how to Learn**“ schließlich ist eine Erweiterung des Transferproblems angesprochen, die schon seit längerem als eine der wichtigsten Basisqualifikationen angesehen wird, um dem Phänomen

rascher technischer und gesellschaftlicher Veränderungen und einer fortlaufenden Verkürzung der „Halbwertszeit des Wissens“ begegnen zu können. Dabei geht es um die Generalisierung von Prinzipien, die Ableitung von Modellen und den Erwerb von Kompetenzen, die den Lernprozeß selbst und die Systematik des damit verbundenen Vorgehens zum Gegenstand haben.

Wie sich „virtuelle“ Lernumgebungen diesbezüglich in geeigneter Weise gestalten und nutzen lassen, ist vermutlich eine der wichtigsten, die Kreativität aller Beteiligten fordernde Frage. Wir nehmen an, daß hierfür insbesondere die Historyfunktion computergestützter interaktiver Systeme - als prozedurales Gedächtnis - systematisch zu einem durch die Lernenden leicht nutzbaren und experimentell veränderbaren Werkzeug (Tool) ausgearbeitet werden müßte.

Neben der Gestaltung und Präsentation des Lernstoffs sind jedenfalls stets auch Gestaltungsansätze gefragt, die der Unterstützung operativer Basisfunktionen dienen. Diese beziehen sich insbesondere auf

- systematisches Explorieren
- Experimentieren als Lernen am (komplexen) Modell mit systematischer Parameter-Variation
- Perspective Taking (Wahrnehmung) und
- Role Taking (soziale Perspektive).

Es wird zu überprüfen sein, inwieweit die in diesem Handbuch vermittelten Gestaltungsprinzipien schon in ihrer jetzigen Form zur Umsetzung der skizzierten übergeordneten Zielsetzungen beizutragen vermögen und wo sie der Ergänzung bedürfen.

2.2.4 Zusammenfassung

Es sollte deutlich gemacht werden, daß einzelne Gestaltungsmittel zur Entwicklung des Lernraumes und des Lernangebotes einer virtuellen Hochschule nur im Zusammenhang mit übergeordneten Gestaltungszielen und -prinzipien gesehen werden dürfen.

Diese ergeben sich - unter Berücksichtigung der Zielgruppe der Lernenden - im wesentlichen aus der Entscheidung für die prinzipielle Lehr-Lern-Konzeption (Lehr-Lern-Philosophie), aus den zu unterstützenden kognitiven Prozessen und aus einer Reihe verallgemeinerbarer Charakteristika (gewissermaßen „Phasen“) des Lernprozesses.

In Abbildung 2.4 sind die wesentlichen Ansatzpunkte für Gestaltungsmaßnahmen abschließend nunmehr unter dem Gesichtspunkt einer soziotechnischen Systemgestaltung veranschaulicht.

Das soziotechnische System „Virtuelle Hochschule“ als Organisation setzt - vermittelt über die technischen Rahmenbedingungen und Restriktionen einer bestimmten Server-Client-Struktur - den Lernenden Ziele (Lernaufgaben) und erhält als Rückmeldung Resultate (Lernergebnisse), anhand derer sich die Zielerreichung fortlaufend überprüfen und der zwischengeschaltete Prozess

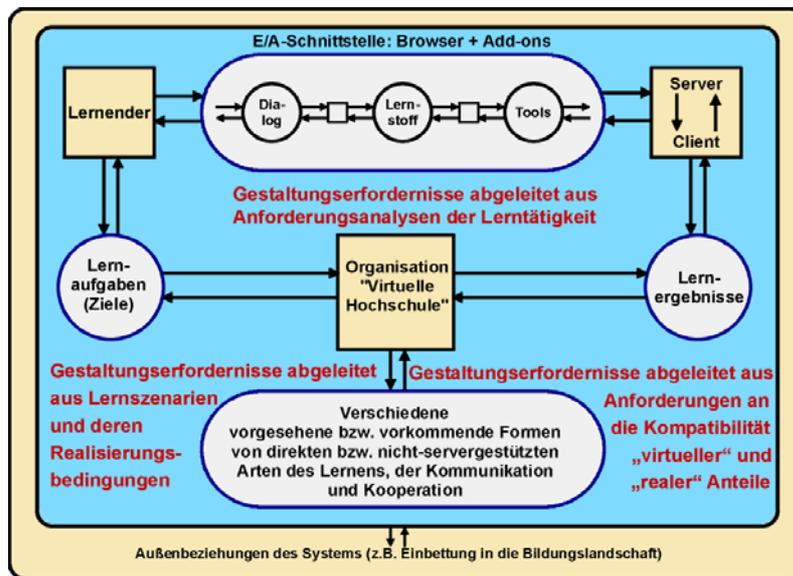


Abbildung 2.4: Gestaltungsschwerpunkte des soziotechnischen Systems „Virtuelle Hochschule“

erforderlichenfalls im Sinne einer Optimierung verändern läßt (obere Hälfte der Abbildung).

Die wesentlichen Gestaltungserfordernisse sind aus Anforderungsanalysen der Lerntätigkeit abzuleiten und beziehen sich auf

- die Gestaltung des eigentlichen Lernstoffs
- die Gestaltung des „Dialogs“ mit dem System (auch als „Navigation“ charakterisierbar) und
- die Entwicklung und Gestaltung von Werkzeugen (Tools) zur Bearbeitung des Lernstoffs, sowie
- eine prinzipielle Ausrichtung aller Gestaltungsansätze an dem Ziel, die Kontrolle des Lernprozesses weitgehend in die Hände der Lernenden selbst zu geben.

Es ist weiter (untere Hälfte der Abbildung) davon auszugehen, daß das System auch über eine Reihe von Formen des Lernens, der Kommunikation und Kooperation der Beteiligten verfügt, die nicht oder nicht ausschließlich über das Server-Client-System technisch übermittelt werden (z.B. in Präsenzphasen). Sie lassen sich in ihrer jeweiligen Spezifik mittels Lernszenarien und deren Realisierungsbedingungen beschreiben und führen zu weiteren Gestaltungserfordernissen, die u.a. auch die Kompatibilität von „virtuellen“ und „realen“ Anteilen dieser Art des Studiums betreffen.

3 Grundzüge eines methodischen Rahmenkonzepts zur Erstellung und Überprüfung von Lehreinheiten

Die Erstellung von Systemen jeglicher Art ist ein Prozess. Dieser Prozess umfasst verschiedene Phasen, abschließende Dokumente oder Ergebnisse und Entscheidungspunkte. Er unterscheidet sich je nach zu gestaltendem System und den verfügbaren Ressourcen. Das vorliegende Handbuch beschreibt einen Prozess zur ergonomischen Gestaltung von Lehreinheiten (bzw. „Modulen“) für eine virtuelle Fachhochschule. Dieser Prozess lehnt sich zum Einen an Erkenntnisse aus dem Softwareengineering und zum Anderen an bereits vorliegende Beobachtungen aus der bisherigen Herstellungspraxis solcher Einheiten an.

Das Ziel der folgenden Beschreibung ist es, Entwickler in die Lage zu versetzen, die Erstellung einer Lehreinheit zu planen. Die Planung setzt sich dabei zusammen aus der Aufgabenverteilung, dem Phasenmodell und der Beschreibung konkreter Maßnahmen zur Sicherstellung der Phasenziele. Dies schließt die Sicht der *Qualitätssicherung*¹ mit ein.

3.1 Kurze Beschreibung des Prozessmodells

Der Prozessablauf (siehe auch Abbildung 3.1 auf Seite 24) kann in sechs verschiedene Phasen unterteilt werden, die hier zunächst nur kurz aufgezählt und dann in den nachfolgenden Kapiteln genauer erläutert werden:

3.1.1 Die Prozessphasen im Überblick

Phase 1: Analyse

Bevor man etwas entwirft, muss man sich über die Ziele und Bedingungen klar werden. Um im Sinne der späteren Zielgruppe und ihrer Wünsche und Bedürfnisse entwickeln zu können, muss man diese zunächst einmal analysieren und die Ergebnisse dokumentieren. Hinzu kommen weitere Anforderungen, die sich zum Beispiel aus den organisatorischen Bedingungen einer virtuellen Hochschule ergeben.

¹Wobei die Zielrichtung dieses Handbuches die *ergonomische* Qualität fokussiert, auch wenn dies implizit im Sinne der Aufgabenangemessenheit auch wiederum die Gesamtqualität beinhaltet.

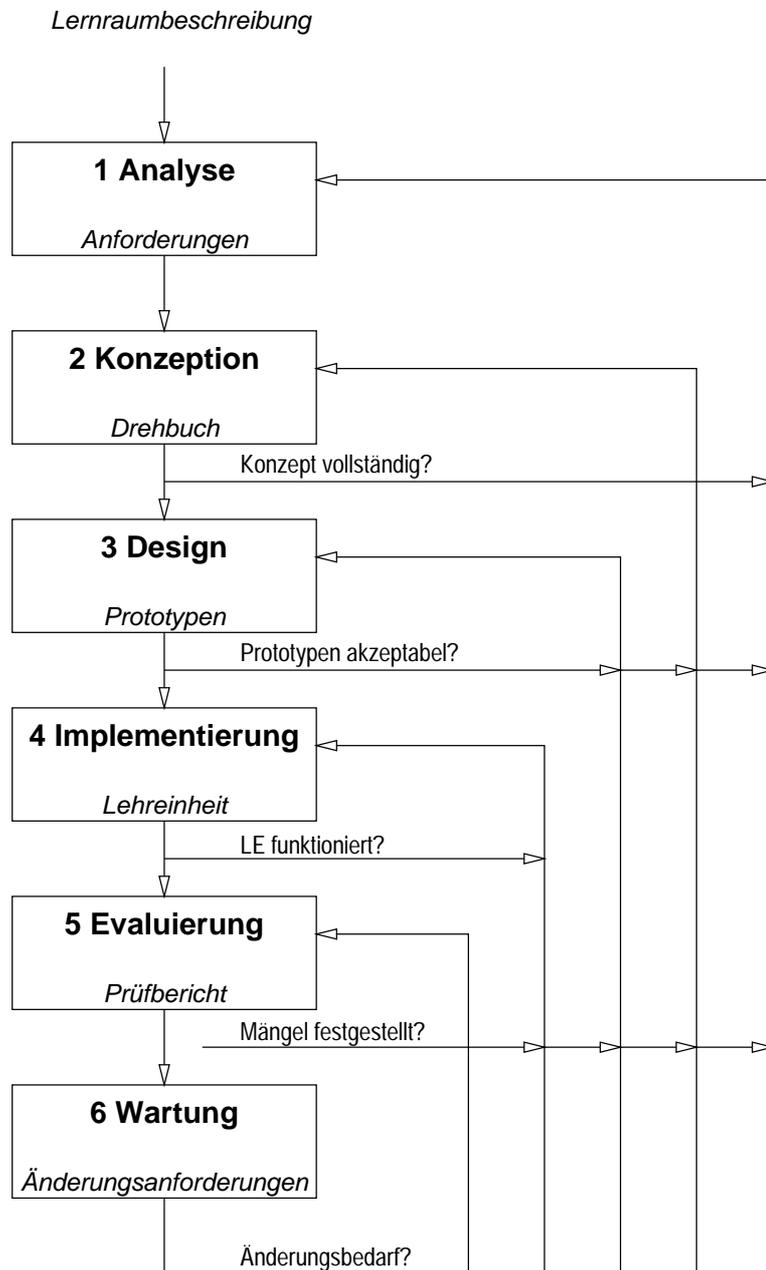


Abbildung 3.1: Die verschiedenen Prozessphasen und die möglichen Rücksprünge zu vorherigen Phasen

Die in dieser Phase entworfenen Dokumente sind die Basis für alle folgenden Schritte und müssen dementsprechend sorgfältig bearbeitet und gepflegt werden. Ausserdem muss sichergestellt werden, dass die Dokumente auch an die beteiligten Personen (Entwickler, Planer aber auch Benutzer) verteilt und von diesen eingesetzt werden, statt in Aktenschränken zu verstauben.

Änderungen in dieser Phase sind einfach (auch im Sinne des Aufwandes) durchzuführen, wohingegen Änderungen in der Implementierung um ein vielfaches „teurer“ sind. Allein schon aus diesem Grunde kann an dieser Stelle mittelfristig ein Großteil des späteren (Änderungs-) Aufwandes vermieden werden.

Das Ergebnisdokument dieser Phase besteht aus einer Beschreibung der gesammelten Daten und den daraus abgeleiteten Anforderungen an die Lernereinheit.

Phase 2: Konzeption

Nachdem so weit wie möglich alle Basisinformationen gesammelt und dokumentiert wurden, müssen diese nun in einem weiteren Schritt in ein Konzept umgesetzt werden. Auch hier besteht die Hauptarbeit aus der Auswahl eines pädagogischen-didaktischen Konzeptes.

Das Ergebnis ist ein *Drehbuch*, welches schon den geplanten Ablauf, nicht aber die technische Realisierung enthält. Beispielsweise würde ein Drehbuch zwar die Szenen eines interaktiven Rollenspieles enthalten, nicht aber bereits die genaue Darstellung der dort verwendeten Buttons oder Grafiken.

Phase 3: Design

Viele Projekte beginnen direkt in dieser (oder gar in der nächsten) Phase, da man irrtümlich annimmt, so den Aufwand der ersten beiden Phasen einsparen zu können. Doch erst *nach* Klärung des Nutzungskontextes (Analyse) und auf Basis eines Konzeptes kann Design mehr sein als nur ein „schickes Layout“ im Sinne einer rein ästhetischen Optimierung.

Das Drehbuch aus der vorhergehenden Phase wird nun „nach allen Regeln der Kunst“ umgesetzt. D.h. dass hier nun die ergonomischen Grundregeln (siehe auch Kapitel 4 auf Seite 43) aber auch designerisches Geschick und ein Auge für ansprechende und angemessene Gestaltung notwendig sind. Sofern vorhanden sind Styleguides und andere Richtlinien zu beachten.

Am Ende des Designs entstehen *Prototypen*, anhand derer das Entwicklungsteam und potentielle spätere Benutzer einen ersten Eindruck von dem zu erstellenden System bekommen können.

Hier lassen sich erste kleine Tests durchführen. So können schon jetzt eventuelle (Design-) Fehlentscheidungen aufgedeckt und einfach behoben werden. Häufig ist es vorteilhaft, zunächst parallel mehrere rudimentäre Prototypen zu erstellen und diese miteinander zu vergleichen.

Auch hier gilt, dass eine frühe Änderung späteren Aufwand vermeidet, der erfahrungsgemäß um den Faktor 10 pro Phase (!) ansteigt. D.h. dass Fehler,

die erst hier statt bereits in der Analysephase behoben werden, bereits 100fach „teurer“ sind.

Zur Veranschaulichung: Vergleichen Sie einmal den Aufwand zur Änderung des Satzes: „Die Benutzer können keine Audiodaten verwenden, da sie mit mehreren Personen in einem Büro arbeiten“ in einem Analysebericht, mit dem Änderungsaufwand, eine ehemals audiolastige Version in eine rein textbasierte Einheit zu ändern. Die Erfahrung in Softwareprojekten hat gezeigt, dass solche „teuren“ Änderungen nicht die Ausnahme sind.

Phase 4: Implementierung

In dieser Phase wird nun die eigentliche Realisierung der Lehreinheit oder des Moduls vorgenommen. Analog zur vorhergehenden Phase gibt es auch Projekte, die direkt hier erst einsteigen. Diese Vorgehensweise wird zu recht als „Quick & Dirty“ tituliert.

Während der Implementierung ist darauf zu achten, dass die Vorgaben aus den vorhergehenden Phasen nicht leichtfertig übergangen oder abgeändert werden. In der Praxis zeigt sich zwar, dass nicht immer alle Planungen tatsächlich so wie ursprünglich vorgesehen umzusetzen sind, doch dies ist dann kein Grund, an dieser Stelle zu „flicken“ oder „mal eben“ etwas zu realisieren.

Die bisherigen Prozessphasen müssen dann mit Blick auf die offenen Fragen noch einmal durchlaufen werden.

Phase 5: Evaluierung

Die Erfahrung zeigt, dass es selten gelingt, auf Anhieb ein perfektes System zu schaffen. Im Bereich der Analyse und der Konzepte bleiben in der Regel häufig einige Fragen offen, die mit vertretbarem Aufwand im Vorfeld nicht zufriedenstellend zu klären sind. Zusätzlich gibt es gelegentlich Missverständnisse und Fehler während des Herstellungsprozesses. Die Aufgabe der Evaluation ist es, solche mit großer Wahrscheinlichkeit enthaltenen Mängel offenzulegen und die Gesamtqualität des geschaffenen Systems zu bewerten.

Die Evaluierung von Lehreinheiten ist insbesondere wichtig, um ein durchgängiges Qualitätsniveau zu erreichen und zu halten. Wichtige Qualitätsaspekte sind

- die Effektivität, hier als pädagogische Eignung („Wird das Lernziel erreicht?“),
- die Effizienz („Ist der notwendige Aufwand angemessen?“) und natürlich
- die Zufriedenheit des Benutzers („Wird das virtuelle Studienangebot akzeptiert und stellt es seine Studenten zufrieden?“).

Phase 6: Wartung

Eine Evaluation macht nur Sinn, wenn festgestellte Mängel auch zu Veränderungen am System führen. Man spricht hier von einem „Lebenszyklus“, der auch die Anpassung an evtl. veränderte Bedingungen und Inhalte mit einschließt. Ein System „lebt“, so lange es auch weiterhin verbessert und angepasst wird. Wartung meint also nicht nur die bloße Pflege und (technische) Optimierung und Instandhaltung eines eingefrorenen Zustandes, sondern auch und gerade die kontinuierliche Weiterentwicklung auf Basis von sich verändernden Benutzereigenschaften und inhaltlichen Anforderungen.

3.1.2 Iterative Entwicklung

Die oben beschriebenen Phasen sind nicht so streng sequentiell und vor allem nicht sofort abschliessend zu bearbeiten. D.h. in einem ersten Anlauf wird man erfahrungsgemäß Defizite bei der Analyse aber auch in den folgenden Phasen in Kauf nehmen, da noch nicht alle benötigten Daten und Informationen zur Verfügung stehen. Wenn diese dann, zum Beispiel durch Erfahrungen aus dem Einsatz und Ergebnissen der Evaluation, später bereit stehen, sollten die vorhergehenden Phasen erneut durchlaufen werden. Theoretisch sollte dies so lange wiederholt werden, bis alle Mängel behoben und eine optimale Form gefunden wurde. In der Praxis werden aber Zeit- und Mittelbegrenzungen die mögliche Anzahl der Iterationen (Durchläufe) dieses Prozesses begrenzen. Mehr als nur *ein* einziger Durchlauf von der Analyse bis zur Evaluation ist aber auf jeden Fall vorzusehen, da ansonsten die Ergebnisse der Evaluation schon nicht mehr sinnvoll in das System eingepflegt werden können.

3.1.3 Mögliche Aufgabenteilung

Die dargestellten Phasen implizieren bereits eine mögliche Aufgabenteilung. Je nach verfügbaren Ressourcen macht es Sinn, die einzelnen Pakete auf verschiedene Personen aufzuteilen. Bei der Erstellung von Lerneinheiten sind verschiedene „Rollen“ zu berücksichtigen, die aber nicht notwendigerweise von jeweils unterschiedlichen Personen wahrgenommen werden müssen. In Tabelle 3.1 ist eine mögliche Aufgabenteilung beschrieben, wie sie in der Praxis häufig anzutreffen ist. Die verschiedenen Phasen und Aufgaben sind in der Praxis nicht scharf voneinander zu trennen, so dass von einer durchgehenden Kommunikation und Kooperation der Beteiligten auszugehen ist. Die Tabelle beschränkt sich auf die Darstellung der jeweiligen Arbeitsschwerpunkte².

Fachautoren

Der eigentliche Inhalt, die „Nutzlast“ einer Lerneinheit, wird von dafür qualifizierten Autoren zusammengestellt. Diese legen auch die Lernziele fest, die

²Die Zusammenfassung von Designer und Ergonom ist derzeit eine Idealvorstellung, denn es wäre wünschenswert, wenn die Grundbegriffe und vor allem Methoden der Ergonomie Teil dieser Rolle wären. In der Praxis trifft dies allerdings nicht immer zu, so dass zur Zeit noch auf weitere externe Kräfte zurückgegriffen werden muss.

3 Methodisches Rahmenkonzept

| Phase/Rolle | Fachautor | Pädagoge | Designer und Ergonom | MM-Producer |
|------------------------------|--|--------------------------------|---|----------------------------------|
| Analyse | Lernziele | Lerner- besonder- heiten | Gestaltungs- anforderun- gen | technische Anforderun- gen |
| Konzeption | Anpassung an den Inhalt | Lehrkonzept | — | — |
| Design | — | — | Umsetzung des Konzep- tes | — |
| Implemen- tierung | — | — | — | Realisierung der Einheit |
| Evaluation | Angemessen- heit bzgl. des Inhalts | didaktische Eignung | gestalterische und ergo- nomische Qualität | technische Qualität |
| Wartung | Änderungsvorschläge | | | |

Tabelle 3.1: Die verschiedenen Rollen und Phasen bei der Erstellung einer Lerneinheit.

mit der Lehreinheit erreicht werden sollen. Die Lernziele sind zumeist organisatorisch, z.B. durch ein Curriculum, vorgegeben. Weitere Einflußfaktoren sind aus der Analysephase abzuleiten, zum Beispiel besondere Lernanforderungen einer virtuellen Hochschule.

In der Konzeptionsphase arbeitet der Fachautor eng mit den Pädagogen (siehe nächsten Abschnitt) zusammen, um so seine Vorstellungen der Stoffvermittlung optimal auf die neuen Möglichkeiten der *virtuellen* Fachhochschule anzupassen.

Letztendlich zeichnet der Fachautor auch meist verantwortlich für die gesamte Lehreinheit und wird auch deshalb bei der Evaluation der Einheit eingebunden, um die angemessene Darstellung „seines“ Themas sicherzustellen bzw. zu überprüfen.

Pädagogen/Didaktiker

Mit „Pädagogen“ seien in diesem Zusammenhang die Tätigkeitsfelder rund um Pädagogik und Didaktik beschrieben. In der Analysephase werden für die zu erwartenden Lerner spezifische Eigenschaften erfasst, die dann als Entscheidungsgrundlage für das später einzusetzende Vermittlungskonzept dienen. In der Evaluationsphase wird der Lehr- bzw. Lernerfolg als Maß angenommen und geprüft.

Diese Rolle wird häufig vom Fachautor mit übernommen und nur durch punktuelle pädagogisch-didaktische Beratung bezüglich der neuen Medien und Möglichkeiten ergänzt.

Designer/Ergonomen

Die aufgaben- und benutzerangemessene und dabei ästhetisch ansprechende Umsetzung des Konzeptes ist die Aufgabe der hier als „Designer“ bezeichneten Rolle. Als Basis werden weitere Benutzereigenschaften aus der Analysephase benötigt. Zum Beispiel die allgemeinen Medienkompetenzen der Benutzer oder das Einsatzumfeld der Lerneinheit. Zu den Aufgaben des Designers gehört eigentlich auch und besonders die Berücksichtigung ergonomischer Grundsätze bei der Gestaltung der Einheit. Da diese Qualifikation derzeit aber nicht immer als vorhanden vorausgesetzt werden kann, setzt hier zumeist die Beratung durch spezialisierte Ergonomen an.

Multimedia Producer

Die eigentliche Realisierung der Einheit ist Aufgabe des Multimedia-Producers (im Folgenden MM-Producer genannt). Die in den vorangegangenen Phasen erarbeiteten Konzepte müssen von ihnen technisch realisiert werden.

Es hat sich als unerlässlich erwiesen, dass Designer und MM-Producer sehr eng zusammenarbeiten (wenn sie nicht sowieso beide Funktionen gleichzeitig innehaben), damit bereits in der Konzeptphase die Machbarkeit innerhalb des Projektrahmens nicht aus dem Blick verloren wird. Ein Konzept, welches mit den vorhandene Ressourcen nicht real umsetzbar ist, ist nutzlos³.

3.2 Beschreibung einer möglichen Vorgehensweise zur Erstellung von Lerneinheiten

Nachdem im vorherigen Kapitel zunächst eine bewusst eher skizzenhafte Betrachtung der Phasen und der beteiligten Rollen stattgefunden hat, werden in den folgenden Abschnitten mögliche Konkretisierungen der notwendigen Massnahmen und mögliche Fehlerquellen beschrieben.

3.2.1 Der Lernraum: Analyse der Lehrziele und des Nutzungskontextes

Wie bereits mehrfach angesprochen ist das Merkmal einer „guten“ Lerneinheit, dass die Zielgruppe später gut damit lernen kann. Dies klingt zunächst trivial, bedingt aber, dass alle dafür relevanten Informationen so genau wie möglich und nötig zusammengetragen werden. Nur wenn man weiß, *wer* die Lerneinheit später benutzt, kann man sich auf sie/ihn einstellen. Zusätzlich sind aber auch andere Rahmenbedingungen zu berücksichtigen, sofern sie die spätere Gestaltung beeinflussen.

Beispielsweise ist es essentiell zu wissen, inwiefern der Benutzer mit dem Medium Computer vertraut ist oder wie häufig und wie lange er voraussicht-

³Zur Ergänzung und Vertiefung sei der Leser auf Kap. 6 der Studie Forschergruppe SofTEC NRW hingewiesen [Nagl et al. 99] Seite 113ff.

lich an einer Lerneinheit arbeiten wird. Nur darauf aufbauend kann sinnvoll entschieden werden, welche vorhandenen Elemente der Benutzungsschnittstelle⁴ (GUI-Controls wie Buttons, Check-Boxen etc.) an welcher Stelle eingesetzt werden können. Auch nicht so naheliegende Informationen können entscheidenden Einfluss auf die Gestaltung nehmen. Zum Beispiel die Arbeitsumgebung, in der eine Lerneinheit benutzt wird, denn dabei entscheidet sich u.a., welche Konzentration und Motivation beim Benutzer zu erwarten ist.

Die ganzheitliche Sicht auf die Umgebung, die Ziele und die Fähigkeiten des Lerners wie auch die angebotenen Informationen und Werkzeuge, bezeichnen wir als **Lernraum**⁵.

Die Optimierung der Eignung (sprich: Gebrauchstauglichkeit oder Usability) eines Lern-Angebotes, sowohl des didaktischen Konzeptes als auch der ergonomischen Gestaltung der Implementierung, ist das Ziel des hier beschriebenen Prozesses. Er ist angelehnt an die Beschreibung eines „aufgaben- und benutzerzentrierten Entwicklungsprozesses“ (vgl. den gleichnamigen ISO Standard [ISO 13407]) aus dem Bereich der Softwareentwicklung und lässt sich hier gut anwenden.

Wo und wie bekomme ich Informationen über den Lernraum?

Zunächst einmal sollten die Auftraggeber und die Autoren der Lerneinheiten eine Definition ihrer Zielgruppe finden und dokumentieren. Im Kontext eines virtuellen Studiums kann man typischerweise im wesentlichen zwischen zwei Hauptgruppen unterscheiden:

- Grundständige Studenten: Vollzeitstudierende, die analog zu „normalen“ Studenten, ihre gesamte verfügbare (Arbeits-)Zeit für das Studium verwenden können.
- Fortbildungsstudenten: Diese Studierenden sind hauptsächlich in einem Beruf tätig und nutzen die virtuellen Angebote als Möglichkeit zur Weiterqualifikation.

In einem ersten Schritt werden nun die Auftraggeber und die Autoren sich ein eigenes Modell vom Vorgehen und den Arbeitsweisen und -techniken dieser Gruppen machen. Die Praxis zeigt, dass solche Modelle sich erheblich vom späteren tatsächlichen Arbeitsablauf bei der Zielgruppe unterscheiden können. Wird dies zu spät oder gar nicht bemerkt, führt es zu nicht oder nur wenig brauchbaren Entwürfen und Implementierungen. Um diese Gefahr zu vermeiden, ist es unerlässlich, direkt mit Vertretern der betrachteten Gruppen zu arbeiten und sie über ihre Vorgehensweisen, Einstellungen, Motivationen und Kenntnisse zu befragen. Noch besser ist es, Vertreter dieser Gruppen bei einer solchen Lerntätigkeit direkt zu beobachten und diese Beobachtungen

⁴Statt „Benutzungsschnittstelle“ werden häufig auch die Begriffe „Benutzerschnittstelle“ oder „Oberfläche“ verwendet. Gegen diese Begriffe sprechen mögliche Fehldeutungen und die in der Vergangenheit sich daran entzündenden (fruchtlosen) Diskussionen.

⁵Diese Verwendung des Begriffes „Lernraum“ weicht von der meist rein technischen Sicht auf ein Softwaresystem zur Verwaltung von Hypertexteinheiten ab und geht darüber hinaus.

zu dokumentieren. Ein solches Dokument kann zum Beispiel ein sogenanntes **Szenario** sein, in dem man in Erzählform die Beobachtungen beschreibt. Diese Szenarien haben den Vorteil, dass man sie mit den Vertretern der Zielgruppe gemeinsam durchgehen kann und so Unstimmigkeiten frühzeitig erkannt werden können („ökologische Validität“).

Da zunächst teilweise noch nicht genügend Prototypen und Vertreter der Zielgruppen verfügbar sind, muss man sich im Notfall mit „intendierten“ Szenarien behelfen. Statt also Beobachtungen zu dokumentieren, entwerfen Auftraggeber und Autor zusammen ein Szenario, in dem sie beschreiben, wie sie sich die Aufgabenbearbeitung *vorstellen*. Sobald erste Prototypen vorliegen, sollten zunächst sowohl diese Szenarien als auch die Prototypen vor Ort mit den potenziellen Benutzern evaluiert werden.

Was sind wichtige Merkmale?

Die Szenarien geben zunächst nur eine Vorstellung von der Arbeitsweise der potenziellen Benutzer. Entscheidend ist aber, für die Gestaltung relevante Informationen aus diesen Dokumenten zu extrahieren. Für das Lehren und Lernen sind eine Reihe von besonderen Benutzereigenschaften und Vorgehensweisen relevant.

Einige allgemein wichtige Merkmale von Benutzern und Ihren Aufgaben sind im Folgenden beispielhaft aber eventuell unvollständig aufgezählt:

Aufgabeneigenschaften Zunächst sollte die Aufgabe des Lernens so exakt wie möglich beschrieben werden:

- Ziel: Was soll gelernt werden?
- Begründung: Warum soll es gelernt werden?
- Aufgabe: Welche Inhalte sind unverzichtbar, welche weniger wichtig?
- Häufigkeit, Dauer: Wie oft und wie lange wird gelernt?

- Ergebnisse:
 - Wie exakt sollen die Inhalte reproduziert, wie sollen sie verallgemeinert und angewandt werden können?
 - Welche Transferleistungen sind zu erbringen?

Benutzereigenschaften Wie bereits beschrieben sind natürlich auch die Fähigkeiten, Kenntnisse und Eigenheiten der (zunächst potentiellen) Benutzer sehr wichtig.

- Technische Vorkenntnisse: Kenntnisse im Umgang mit der verwendeten Software (hier Browser, ggf. EMail-Programme und zusätzliche Tools) als auch mit dem generellen Look&Feel der Arbeitsoberfläche (meist Windows 9x).
- Motivation des Benutzers: Warum beschäftigt sich der Benutzer mit der angebotenen Lerneinheit? Was will er erreichen⁶?
- Verfügbare Kapazität: Wieviel Lernaufwand für die Bedienung des Lernmodules ist zumutbar? Lerntempo? Stoffmenge?
- Umgebung beim Benutzer: In welcher Umgebung wird der Benutzer mit dem Lernmodul arbeiten? Wird er ungestört und voll konzentriert arbeiten können? Welche Unterbrechungen sind wahrscheinlich und in welchen Intervallen?

Organisatorische Rahmenbedingungen Auch eine Organisation wie die Virtuelle Fachhochschule hat natürlich Vorgaben, die bei der Gestaltung der Einheiten berücksichtigt werden müssen.

- Ästhetisches Design, Moderner Look: Für die Positionierung in der Bildungslandschaft zwischen einer wachsenden Zahl von konkurrierenden Anbietern virtueller Lerneinheiten ist auch die Ästhetik einer Webpräsentation wichtig.
- Zeit- und Arbeitsvorgaben: Wieviel Zeit soll der Studierende typischerweise für die Bearbeitung der Lerneinheiten aufwenden? Welche Arbeitsformen sollen explizit gefördert werden?

Technische Rahmenbedingungen Last but not least sind natürlich auch die einsetzbaren Hilfsmittel (JAVA, ActiveX, Shockwave, Macromedia-Director, Video [MPG vs. AVI], Frames, etc.) wichtige Rahmenbedingungen bei der Erstellung einer Lerneinheit bzw. eines Lehrmoduls. Diese sind aufgrund empirischer Erkenntnisse („Wieviele der Benutzer werden XY ohne Probleme einsetzen können?“) und natürlich organisatorischer Vorgaben („Welche Bandbreite ist von den Studierenden bereitzustellen?“) festzulegen.

⁶Häufig haben die Anbieter einer Einheit eine andere Vorstellung von dieser Motivation als die späteren Benutzer. In diesem Fall ist immer die tatsächliche und nicht die möglicherweise idealisierte Vorstellung der Motivation zu berücksichtigen.

Weitere wichtige Eigenschaften werden vermutlich im Laufe des Projektes entdeckt. Prinzipiell sollte alles dokumentiert werden, was für die Beantwortung der späteren Frage „Erreicht das Lernmodul seinen Zweck?“ hilfreich ist.

Wie werden Ergebnisse der Analyse dokumentiert?

In einem ersten Schritt kann dies völlig informell geschehen, indem man die entsprechenden Merkmale („Die typischen Benutzer haben ca. 1h am Tag Zeit“) gesammelt in einem Textdokument aufschreibt. Dieses Dokument sollte dann allen am Herstellungsprozess beteiligten Personen zur Verfügung stehen.

Insbesondere sollten diese Merkmale, nachdem man sie auch für Laien lesbar aufgeschrieben hat (Klartext statt Codes und Abkürzungen), mit den potenziellen Benutzern noch einmal auf eventuelle Missverständnisse hin untersucht werden.

Da diese Merkmale bei der späteren Untersuchung der Lerneinheiten wieder benötigt werden, ist eine tabellarische Form zweckmäßig (Merkmalsart und Beschreibung des Merkmals). Änderungen und Ergänzungen an diesem Dokument im Verlaufe des Entwicklungsprozesses müssen in geeigneter Form an die Betroffenen weitergeleitet werden. Dieses Dokument muss *lebendig* sein, da es ansonsten seinen Sinn verliert.

3.2.2 Konzeption

Auf Basis der in der ersten Phase bestimmten Merkmale des Lernraumes kann nun ein Konzept erstellt werden. Es ist sinnvoll und fast immer notwendig, zunächst diese Konzeption zu dokumentieren, um so für die verschiedenen am Herstellungsprozess beteiligten Personen eine gemeinsame Entscheidungsplattform zu haben. Für die Konzeption gilt ebenso wie für die Ergebnisse der Analyse, dass sie *lebendig*, d.h. aktuell und konsistent sein muss, um Sinn zu machen. Der kurzfristige Zeitgewinn durch einen direkten Sprung von der Analyse zur Implementierung („ad hoc Programmierung“) wird spätestens bei der Evaluation und bei der Wartung mit einem unangemessen hohen Aufwand bezahlt und rechnet sich ökonomisch nicht.

Eine wichtige Funktion eines Konzeptionsdokumentes ist, dass hier Entscheidungen nicht nur festgelegt, sondern auch die Entscheidungsgründe dokumentiert werden. D.h.: Nachdem in der vorherigen Phase zunächst einmal relevante Daten identifiziert und gesammelt wurden, gilt es hier, die Entscheidungen auf Basis dieser Daten nachvollziehbar zu machen. Dies ist nützlich, wenn man später Abweichungen von hier getroffenen Entscheidungen bewerten will oder im Lebenszyklus eines Lernmoduls Personen und Rollen wechseln und dann zunächst einmal die „alten“ Entscheidungen nachvollzogen werden müssen.

Lernziele bestimmen

Die Eigenschaften der Aufgabe, die in der vorhergehenden Phase analysiert wurden, enthalten bereits implizit die Beschreibung der Lernziele, sowohl aus Sicht der Benutzer als auch der Anbieter. Im Konzept sind diese nun verbindlich festzulegen. Sind Ziele in der Analysephase unklar geblieben, so sollten sie spätestens zu diesem Zeitpunkt geklärt und für die weitere Entwicklung eindeutig dokumentiert werden.

Ableitung besonderer Anforderungen aus dem Nutzungskontext

Ein häufig übersehener Punkt bei der Analyse von Merkmalen des Lernraumes sind **besondere Anforderungen**, d.h. Anforderungen die nicht immer zwingend bereits bei der Aufgabe vorgesehen wurden. Ein Beispiel ist die Arbeitsplatzsituation beim Studierenden. Bestimmte Techniken und Lehrmethoden sind unter Umständen für den Einsatz in der häuslichen Umgebung weniger geeignet, auch wenn sie sich vielleicht von der Aufgabe her anbieten würden. Beispielsweise sind auf Videokonferenzen gestützte Lehrformen aufgrund der zu erwartenden mangelnden Hardware- und Netzwerkleistungen beim Benutzer kritisch zu betrachten, auch wenn diese Systeme zunächst besondere Möglichkeiten bieten.

Als eine andere spezifische Anforderung könnte sich herausstellen, dass die typischen Studierenden erst *am Ende* eines Arbeitstages mit einer Lehrinheit arbeiten und dementsprechend weniger aufnahmefähig sein würden.

Durch die grosse Zahl und die sehr verschiedenen Vorkenntnisse und Einstellungen der beteiligten Lernmodul-Entwickler in verteilten Projekten entstehen schnell inkonsistente Gestaltungen und Präsentationsformen, die den Benutzer unnötig belasten. Die konsistente Gestaltung der Module untereinander aber auch zu den Lernenden bekannten Konzepten ist deshalb eine zentrale Anforderung.

Auswahl des didaktischen Konzeptes

Unter Berücksichtigung der Lernziele und der besonderen Anforderungen aus dem Nutzungskontext ist nun ein didaktisches Konzept zu wählen und verbindlich festzulegen.

Entwurf eines Drehbuches

Als geeignete Dokumentationsform für ein Konzept kann ein erweitertes „Drehbuch“ benutzt werden. Ein solches Drehbuch enthält die Inhalte des Lernmoduls entlang des gewählten didaktischen Konzeptes strukturiert, nicht aber das Design oder gar die Implementierung. Die notwendigen Elemente des Lernmoduls sollten deshalb nur abstrakt beschrieben werden („Button zum Blättern auf die nächste Seite“ statt „Grauer Button `buttonright.gif` oben rechts“). Zusätzlich sollten alle verbindlichen Designentscheidungen aufgezählt werden (Bsp.: „Videos über 5MB sind zulässig, da die Einheit per CD ausgeliefert wird“).

3.2.3 Design

Jetzt erst folgt der Schritt zur tatsächlichen (auch ästhetischen) Ausgestaltung der Lerneinheiten. Die in der ersten und zweiten Phase festgestellten und dokumentierten Anforderungen (z.B. „Welche Bedienelemente wird der Benutzer kennen?“) sind die Basis für die zu treffenden Designentscheidungen. Qualifikationen aus dem Bereich Design und Gestaltung sind in dieser Phase notwendige Voraussetzungen.

Prototyping

Anhand des Drehbuches wird eine erste Roh-Fassung von (Teil-)Ausschnitten der Lerneinheit realisiert. Dabei können auch Werkzeuge angewendet werden, die bei der späteren Implementierung der Einheit nicht mehr benutzt werden. Wichtig ist nur, dass die Benutzungsschnittstelle und die Struktur der Rohfassung der vorgesehenen späteren Implementierung möglichst nahe kommt.

Da es hier noch relativ leicht und vor allem kostengünstig möglich ist Änderungen vorzunehmen, sollte der Prototyp ausgiebig getestet und zusammen mit Benutzern geprüft werden (Abschnitt 3.2.5 Seite 36). Sind Mängel oder Probleme festzustellen, so müssen die letzten Schritte wiederholt werden.

Stellt man beispielsweise fest, dass die hinzugezogenen Benutzer mit der Benutzerführung Probleme haben, so ist zunächst zu prüfen, ob man in der Analysephase nachbessern muss, denn vielleicht hat man ihre Kenntnisse und Fertigkeiten falsch eingeschätzt. Ist dies nicht der Fall, so ist zu überlegen, ob das Konzept didaktisch und ergonomisch für die Zielgruppe geeignet ist. Ist man sich auch hier sicher, so können nun Änderungen am Design vorgenommen werden, die dann in einem neuen Prototypen berücksichtigt werden. Für diesen Prototypen kann es ausreichen, wenn man nur einen Teil neu gestaltet, um festzustellen, ob das nun überarbeitete Konzept besser angenommen wird als die bemängelte Vorgängerversion.

Manchmal scheint die Entscheidung naheliegend, etwas „erst einmal so zu lassen“. Dabei ist aber zu beachten, dass spätere Änderungen meist am dann notwendigen Aufwand scheitern. Echtes Prototyping sieht deshalb sogar vor, dass die Prototypen *gar nicht* weiter verwendet werden. Eine solche strenge Auslegung ist in diesem speziellen Fall vielleicht nicht notwendig, aber in der Prototyping-Phase sollten Änderungen prinzipiell erwogen werden, bevor es zu spät ist.

3.2.4 Implementierung

Nachdem nun ein Prototyp entwickelt und überprüft wurde, werden die eigentlichen Inhalte entsprechend den Vorgaben der Konzeptionsphase und des so gefundenen Designs umgesetzt⁷. Dabei ist es entscheidend, dass nun keine

⁷Auch wenn die Implementierungsphase im Handbuch nur einen kleinen Raum einnimmt, so stellt sie natürlich die wesentliche und arbeitsintensivste Phase des gesamten Projektes dar.

abweichenden konzeptuellen oder designerischen Entscheidungen mehr einfließen. Sollte sich herausstellen, dass man einen bestimmten Fall vergessen hat, so sind für diesen (ggf. im Schnelldurchgang) die vorhergehenden Schritte nachzuholen.

Die große Gefahr bei an dieser Stelle getroffenen abweichenden Entscheidungen ist, dass kleine, sich dadurch einschleichende Mängel das gesamte Produkt unbrauchbar machen oder zumindest den Nutzen und damit die Gebrauchstauglichkeit extrem mindern können. Beispielsweise könnte die nachträgliche Änderung von Schriften und Farben oder das Hinzufügen von Zusatzfunktionen den Wert des bisher entwickelten Konzeptes völlig verändern.

Funktionale Tests

Zur Gebrauchstauglichkeit im engeren Sinne gehört auch die sogenannte „Benutzbarkeit“, d.h. das Funktionieren der angebotenen Elemente und aktiven Inhalte. Bis in die Prototypingphase standen zunächst grundsätzliche Überlegungen im Vordergrund, doch während und nach der Implementierung ist nun auch auf die vollständige und korrekte Umsetzung und Ausprogrammierung der vorgesehenen Inhalte und Funktionen zu achten. Dazu sind funktionale Tests aller Links und der enthaltenen (zum Teil aktiven) Inhalte (z.B. Videos oder JAVA-Applets) unverzichtbar. Aus Sicht der Ergonomie ist zu beachten, dass Benutzer nach technischen Fehlern häufig diese auch auf andere, eigentlich nicht betroffene Programmteile übertragen. Im schlimmsten Fall führt dies dazu, dass bestimmte Inhalte (z.B. Applets) in der Erwartung, dass sie sowieso nicht funktionieren, gemieden werden.

3.2.5 Evaluierung

Nach der Erstellung eines Produktes ist es notwendig, dass der IST-Zustand mit den SOLL-Vorgaben der Analyse- und Konzeptphase verglichen wird. Es wird dabei verglichen, ob und inwieweit die in den Konzepten vorgesehenen Anforderungen und Ziele erreicht werden. Diese *Qualitätssicherung* sollte die folgenden Aspekte betrachten (nach [ISO 9241] Teil 11):

Effektivität: Führt das Produkt zu den gewünschten Ergebnissen?

Das gewünschte Ergebnis einer virtuellen Fachhochschule ist primär der Lernerfolg, d.h. die Vermittlung von Inhalten und Fertigkeiten. Eine Evaluation der Effektivität ist also eine Überprüfung, ob die Lernenden die definierten Lernziele erreicht haben. An dieser Stelle sind die in der Analyse und Konzeption dokumentierten Lernziele zu betrachten.

Sinnvolle konzeptionelle Vorarbeit kann an dieser Stelle aber bereits erhebliche Einsparungen bringen.

Effizienz: Ist der notwendige Aufwand sinnvoll und akzeptabel?

Der Einsatz von zusätzlichen oder neuen Werkzeugen muss sich immer auch gegenüber bereits bestehenden Möglichkeiten oder Werkzeugen messen lassen. Es muss sichergestellt werden, dass durch die neuen Möglichkeiten einer virtuellen, computergestützten Lehr-/Lernform nicht unnötiger Aufwand bei den (potentiellen) Benutzern entsteht.

Im konkreten Fall muss ein Studium an einer virtuellen Fachhochschule eine Verbesserung (also auch Erleichterung) gegenüber konkurrierenden Ansätzen (zum Beispiel Fern-Universitäten bzw. Fern-Fachhochschulen) und anderen Möglichkeiten (z.B. „klassisches“ auf gedruckte Dokumente gestütztes Selbststudium) darstellen.

Bei der Aufwandsbetrachtung ist zu beachten, dass es im Lernprozess verschiedene Aufwandsarten und verschiedene Arbeitsphasen gibt. In Tabelle 3.2 ist eine mögliche Strukturierung angegeben. Es ist nun zu untersuchen, inwieweit der Aufwand der einzelnen Phasen noch durch den Nutzen (siehe Effektivität) gerechtfertigt ist. Wenn nicht, so ist entweder der Nutzen zu erhöhen oder Aufwand zu senken ⁸.

Zufriedenheit der Benutzer

Die Annahme, dass die Optimierung der Effektivität und der Effizienz alleine ausreichen und zu zufriedenen Benutzern führen würde, hat sich in der Praxis nicht bestätigt. Zwar führen ineffektive und/oder ineffiziente Lösungen meist zu einer beeinträchtigten Zufriedenheit der Benutzer, doch umgekehrt gilt nicht, dass effektive und effiziente Systeme automatisch die Zufriedenheit erhöhen. Zufriedenheit ist ein höchst subjektives Empfinden des Betroffenen und kaum objektiv zu messen. Trotz alledem ist es entscheidend und zum Beispiel auch für Arbeitsplatzsoftware zwingend gesetzlich vorgeschrieben, dass die Zufriedenheit der Benutzer bei der Bewertung der Gebrauchstauglichkeit mit einbezogen wird.

Die möglichen Ursachen für Unzufriedenheit mit einem System im allgemeinen und einem Lernmodul im besonderen können vielfältig sein, deshalb hier nur ein paar Beispiele:

- **Empfundene Ineffizienz und/oder Ineffektivität**

Analog zu der bereits evaluierten Effizienz und Effektivität empfindet auch der Benutzer selbst ein Kosten-Nutzen-Verhältnis beim Einsatz der Lerneinheit. Dieses kann subjektiv schlechter als das tatsächlich gemessene Verhältnis ausfallen, weil es der Benutzer zum Beispiel belastender findet an einem Rechner zu sitzen als die doppelte Zeit an einem Buch zu verbringen („Früher ging das alles einfacher und schneller“).

Es kommt häufig vor, dass man die Effizienz zu optimistisch beurteilt hat, weil man sich zum Beispiel nur auf die Messung von Bearbeitungs-

⁸An dieser Stelle sei das mahnende Beispiel der Büroautomatisierung genannt, die die Produktivität von Büros nach verschiedenen Untersuchungen bisher in viel geringerem Maße gesteigert hat, als man ursprünglich annahm.

3 Methodisches Rahmenkonzept

| Phase/Aufwandsart | Beschreibung | Zeitaufwand | Lernaufwand |
|---------------------|--|--|---|
| Einrichtung | Einrichtung des Arbeitsplatzes | Beschaffung und Einrichtung der Hard-/Software und des Arbeitsplatzes; Anmeldeformalitäten | Bedienung des Werkzeuges Computer und der verwendeten Software (Browser, techn. Lernraum, Kommunikationstools) |
| Organisation | Der Lernende macht sich einen eigenen Plan, wann und wie er studieren will | Durchsicht der angebotenen Inhalte, Abschätzung des Lernaufwandes | Bewertung der vorgegebenen und der eigenen Lernziele; Abschätzung des eigenen Lernverhaltens und Lern tempos; Auswahl der Lernmethode |
| Lernphase | Be- und Verarbeitung der angebotenen Inhalte | Aufwand zur Aufnahme und Bearbeitung der Inhalte | Aufwand der durch das Lehrkonzept des Lehrers und das Lernkonzept des Lerner und deren Kompatibilität entsteht. |
| Abschluss | Beendigung der Bearbeitung und Überprüfung des Lernerfolges, zum Beispiel durch vorgeschriebene Prüfungen, aber auch durch Selbstprüfung | Vergleich von SOLL-Vorgaben und tatsächlichem Wissen | Finden eines geeigneten Vergleichskonzeptes bzw. einer Methode zur Selbsteinschätzung. |
| Archivierung | Bereitstellung der bereits gelernten Inhalte für eventuelle spätere Zugriffe nach Abschluss der Einheit | Organisation eines (eigenen) Archives | Konzeptbildung und Strukturierung des angeeigneten Wissens zum späteren Wiederzugriff |

Tabelle 3.2: Beispiel für verschiedene Aufwandsarten und mögliche Arbeitsabschnitte bei der Nutzung des virtuellen Studienangebotes bezogen auf ein einzelnes Lernmodul

zeiten gestützt hat. Der tatsächliche mentale Aufwand oder die Belastung beim Benutzer kann in der Praxis aber darüber liegen (zum Beispiel weil der Arbeitsplatz schlecht eingerichtet und mit ungeeigneter Hardware versehen wurde).

- **Schlechte (unästhetische) Gestaltung**

Benutzer haben eine Erwartungshaltung, die nicht immer ausschließlich funktional und rational geprägt ist. Auch Merkmale wie das „gute Aussehen“ einer Präsentation können die Gesamtbeurteilung durch den Benutzer wesentlich beeinflussen.

- **Inhaltliche, organisatorische und außerhalb der Lerneinheit anzuesiedelnde Probleme**

In Arbeitsplatzsituationen kann man beobachten, dass die Zufriedenheit der Benutzer trotz augenscheinlich gut geeigneter Software eingeschränkt ist. In solchen Fällen sind häufig Einflußfaktoren außerhalb des eigentlichen Lernmoduls beteiligt, wie zum Beispiel das Umfeld oder organisatorische Probleme. Auch durch den Lernstoff implizierte Überforderung des Lernenden kann als sich als allgemeine Unzufriedenheit mit den angebotenen Werkzeugen und Medien äußern.

In solchen Fällen sind diese Problemfaktoren zu identifizieren und ihre Behebung anzustreben, denn auch sie können den Nützlichkeitswert eines Lernmoduls konterkarieren.

Evaluationsansätze

Zur Feststellung der Übereinstimmung von IST- und SOLL-Zustand stehen verschiedene Ansätze zur Verfügung. Diese Ansätze unterscheiden sich in ihrem Aufwand, dem Zeitpunkt ihrer Anwendung und ihrer Zielsetzung erheblich. Bei der Entwicklung eines Studienmoduls oder einer Lerneinheit ist es in der Regel erforderlich, alle drei der im folgenden kurz beschriebenen Ansätze einzusetzen, um sämtliche genannten Aspekte der Gebrauchstauglichkeit eines Moduls zu prüfen:

- **Begutachtung**

Die einfachste, schnellste und naheliegendste Möglichkeit der Qualitätssicherung ist die Begutachtung. Eine Möglichkeit ist die *Expertenbegutachtung (Expert Review)*: Spezialisten der Software-Ergonomie unterziehen das Modul einer kritischen Durchsicht und identifizieren Aspekte, die ihrer Erfahrung nach Gebrauchstauglichkeitsprobleme mit sich bringen können. Der Vorteil dieser Methode ist, dass sie sehr schnell ist, schon nach ein bis zwei Tagen hat man in der Regel ein erstes Ergebnis. Insbesondere Erfahrungen aus anderen Bereichen, aber auch aus der Begutachtung anderer Module können hier gut genutzt werden.

Der Nachteil dieser Methode ist, dass sie nur eine vorläufige Abschätzung potentieller Probleme liefert. Untersuchungen zeigen, dass dabei immer noch erhebliche Gebrauchstauglichkeitsprobleme übersehen werden können. Tatsächliche Benutzer finden unter

Umständen überraschend andere Bedienkonzepte und ganz eigene Probleme. Es ist schlicht unmöglich, die tatsächlichen Gegebenheiten bei den späteren Benutzern genau vorauszuahnen, da diese von viel zu vielen meist unbekanntem Parametern abhängen.

Die noch einfachere Variante ist die Begutachtung durch direkt Beteiligte, wie etwa die Autoren und Entwickler / Designer einer Lerneinheit, auf Basis z.B. einer Checkliste (siehe z.B. Kapitel 5). Diese schwächste Form der Begutachtung hat gegenüber dem Expertenreview den Vorteil, dass sie ohne Zuhilfenahme externer Ressourcen auskommt. Der Nachteil ist aber, dass sie meist die auch aus anderen Bereichen bekannten Probleme der „Betriebsblindheit“ in sich birgt und Probleme, die auf Vorlieben und / oder unzutreffenden Annahmen über die künftigen Benutzer beruhen, in der Regel unentdeckt bleiben.

Als Konsequenz sollte die Eigenbegutachtung zwar im Projektverlauf immer wieder stattfinden, doch auf die Hinzunahme von Experten - zumindest in grösseren Abständen - sollte keinesfalls verzichtet werden. Dabei gilt, wie bisher auch: je früher Probleme entdeckt werden, desto einfacher (billiger) ist es, sie zu beheben. Es ist aber falsch, sich allein auf solche Begutachtungen zu verlassen, sie wirken quasi nur als erste „Grobfilter“.

- **Benutzertests**

Benutzertests bilden den unverzichtbaren Bestandteil einer effizienten Evaluation. Statt sich spekulativ in die Rolle eines potentiellen Benutzers zu versetzen und zu versuchen, sich dessen mögliche Vorgehensweisen und Vorkenntnisse vorzustellen, beobachtet man hier repräsentative Vertreter der späteren Benutzer bei der tatsächlichen Arbeit mit dem System. Nachteil dieses Vorgehens ist der notwendige Aufwand bei Durchführung und Auswertung. Doch auch dieser Aufwand rechnet sich, verglichen mit der Qualität der so gewonnenen Ergebnisse.

Benutzertests sollten so früh wie möglich durchgeführt werden, d.h. praktisch sobald ein lauffähiger Prototyp bereit steht. Dabei ist zu beachten, dass es sich bei einem Studienmodul um ein System handelt, welches mittel- oder langfristig verwendet wird. Es ist deshalb eine gewisse Einarbeitungsphase bei der Benutzung des Systems vorzusehen. Praktisch können solche Benutzertests mit Pilotmodulen durchgeführt werden. Eine Kombination mit Fragebögen (siehe den folgenden Absatz) komplettiert das Evaluationskonzept.

- **Fragebögen**

Eine weitere sehr ökonomische Methode zur Feststellung insbesondere von subjektiven Einschätzungen (z.B. der Zufriedenheit) stellen Fragebögen dar. Vorteile von Fragebögen sind ihre einfache Verteilung und der relativ geringe Aufwand beim Befragten. Als Nachteil wird verschiedentlich die Gefahr geringer „Objektivität“ gesehen. Dies gilt aber vor allem dann, wenn darauf verzichtet wird (oder keine Möglichkeit dazu

besteht), die Befragungsergebnisse korrelativ zu denen anderer Untersuchungsverfahren in Beziehung zu setzen. Zu den bekannten Problemen gehören die Tendenz zur erwartungskonformen Beantwortung von Fragebögen („Was wollen Sie von mir hören?“) und zur Mitte („Ist irgendwie OK.“) sowie zum Einfluß verbreiteter Stereotypen. Schließlich gibt es auch soziale und organisatorische Parameter, die die Beantwortung der Fragen beeinflussen können („Wird das später für/gegen mich verwendet?“; „Wie anonym ist die Befragung wirklich?“).

Um diesen Problemen zu begegnen, ist die Konstruktion und Auswertung der Fragebögen nach testtheoretischen Gesichtspunkten sehr wichtig. Ein naives Herangehen („Wie fanden Sie dieses Studienmodul? — Gut/OK/Schlecht“) ist meist nicht aussagekräftig.

Fragebögen eignen sich gut als Ergänzung und Vervollständigung der bereits genannten Methoden der Expertenbegutachtung und der Benutzertests. Für sich alleine sind sie immerhin ein guter Indikator, wie die Qualität des Studienmoduls subjektiv vom Benutzer wahrgenommen wird (vgl. dazu auch die Ausführungen zur „Zufriedenheit“ (siehe auch Kapitel 3.2.5 auf Seite 37)).

Zur Durchführung von Benutzerbefragungen zur Evaluation von Lehrmodulen bzw. -einheiten steht ein Fragebogen zur Verfügung [Triebe 2001], der sich jederzeit mit Studierenden einsetzen lässt.

Wesentliche Gestaltungsaspekte sind dabei sowohl hinsichtlich ihrer prinzipiellen Wichtigkeit zu beurteilen, als auch bezüglich der Zufriedenheit einzuschätzen, die der jeweilige Benutzer (Lernende) aufgrund der angesprochenen gestalterischen Details der jeweiligen Lerneinheit empfand. Die Antwortskalen zur Wichtigkeit und Zufriedenheit sind graphisch unterschiedlich gestaltet, da auf diese Weise einer möglichen Tendenz vorgebeugt wird, beide Aspekte in Abhängigkeit voneinander zu beurteilen.

3.2.6 Wartung von Lehreinheiten

Ein entscheidender Aspekt eines „guten“ Studienmoduls ist, dass es *lebt*. Dazu gehört, dass es über seine gesamte Einsatzzeit (Lebensdauer) hin betreut und aktualisiert wird. Sehr viele Parameter, die die Gebrauchstauglichkeit der Studienmodule betreffen, sind nicht über längere Zeiten hin vollkommen konstant. So ändern sich Lernziele, Benutzereigenschaften und natürlich der Stand der Technik. Daher genügt es nicht, ein Studienmodul zu einem definierten Zeitpunkt fertigzustellen und dann nicht mehr weiter anzupassen. Ähnlich wie die Neuauflagen eines Buches müssen auch Studienmodule aktuell und zeitgemäß gehalten werden.

Im Sinne des hier vorgestellten Prozesses bedeutet dies, dass Studienmodule und die Inhalte von Lerneinheiten immer wieder evaluiert und an die sich ändernden Anforderungen angepaßt werden. Konkret sollte die Wartung zumindest jährlich einmal eine kurze Evaluation, zum Beispiel mittels eines Benutzertests oder einer Fragebogenauswertung am Ende eines Semesters, vor-

sehen. Die dann offenbar werdenden Änderungswünsche sollten den gleichen Prozess durchlaufen wie die gesamte Einheit, d.h. zunächst sollten die bedeutenden Veränderungen identifiziert werden, dann daraus Anforderungen abgeleitet und diese schließlich in einem geänderten Konzept dokumentiert werden.

Um all dies mit vertretbarem Aufwand zu ermöglichen ist es notwendig, die Module von Anfang an „wartbar“ zu halten. Dazu gehören Kommentare innerhalb der erzeugten Dokumente, aber auch die Dokumentation der berücksichtigten Eigenschaften und der daraus abgeleiteten Forderungen und Konzept- bzw. Design-Entscheidungen, wie es der hier vorgestellte Prozess nahelegt.

4 Detaildarstellung grundlegender Regeln

Im praktischen Einsatz sind neben den allgemeinen Konzepten zur Gebrauchstauglichkeit aus den vergangenen Kapiteln auch konkretere Handlungsvorschläge notwendig. Diese können zum einen aus dem Pool des Expertenwissens der Normung (ISO 9241) und zum anderen aus einschlägigen Ratgebern (sehr empfehlenswert [Nielsen 93]) und aus dem Internet gewonnen werden. In diesem Kapitel werden die grundlegendsten Prinzipien kurz dargestellt und anhand eines Beispiels beschrieben.

Die dargestellten Grundregeln sollten im Rahmen der eigentlichen Entwicklung durch die Beteiligten entsprechend übertragen und angewendet werden. Sie sind also nicht im Sinne einer technischen Detailanweisung, sondern mehr als allgemeiner Ratschlag zu sehen und zu verwenden.

4.1 Regeln zur Dialoggestaltung

Die hier dargestellten Regeln sind aus der Norm ISO 9241-10 entnommen. Sie stellen einen Ausschnitt aus den derzeit mehr oder weniger zumindest in Deutschland anerkannten Mindestforderungen an Software dar.

4.1.1 Aufgabenangemessenheit

Definition

Die Software muß die Erledigung der Arbeitsaufgaben unterstützen, ohne durch spezielle Eigenschaften die Benutzer zusätzlich zu belasten. Ausgehend von den auszuführenden Tätigkeiten kann ein Anforderungsprofil an die Software erstellt werden. Soweit sich Arbeitsschritte aus Eigenschaften des Systems ergeben, nicht jedoch aus den Aufgaben der Benutzer, sollten sie auch vom System selbst ausgeführt werden. Die Software soll keine Veränderung der Arbeitsabläufe erfordern, die im Gegensatz zur tätigkeitsbedingten zeitlichen Reihenfolge stehen. Gleichwohl können aber bei organisatorischen Änderungen Arbeitsabläufe überprüft und eventuell verbessert werden. Die von der Software verwendeten Begriffe und Symbole müssen den arbeits-spezifischen Regelungen entsprechen und eindeutig, widerspruchsfrei und möglichst abkürzungsfrei sein. Dies gilt etwa für Funktionsbeschreibungen, Bildschirmmasken, Hilfetexte, sonstige Darstellungen auf dem Bildschirm und für Benutzerhandbücher.

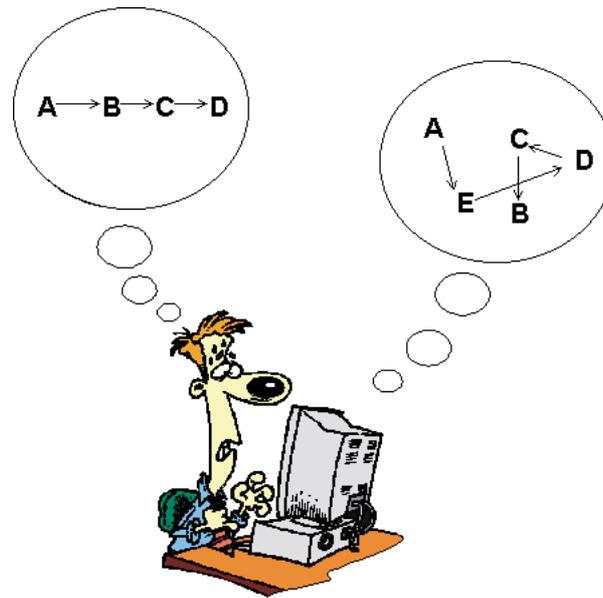


Abbildung 4.1: Unterschiedliche Vorstellungen eines Handlungsablaufes führen zur Belastung des Benutzers.

Beispiel Aufgabenangemessener Dialog

Die in Abbildung 4.1 skizzierte Situation ist typisch für eine nicht aufgaben- (und benutzer-) angemessene Gestaltung. Die Designer des Programms aber auch der Web-Präsentation hatten anscheinend eine andere Vorstellung von der Vorgehensweise als sie nachher tatsächlich beim Benutzer vorzufinden ist. Nach einer Studie der englischen Beratungsfirma *System Concepts* (siehe [IT Management 99]) sind so fast 60% aller Probleme begründet.

Konsequenzen

Fehlende Aufgabenangemessenheit ist wohl nicht Folge des bösen Willens der Entwickler, sondern vielmehr in Versäumnissen in der Analysephase begründet. Wer sich vorher nicht den Aufwand gemacht hat, seine Benutzer und ihre Vorgehensweise zu dokumentieren und zu verstehen, wird mit ziemlicher Sicherheit am Bedarf der späteren Benutzer vorbei entwickeln. Bisher gab es dazu zwei typische Fälle:

1. Die Entwickler waren selbst auch Teil der späteren Benutzergruppe und konnten so von ihrer eigenen auf die Vorgehensweise anderer schließen. So entstehen z.B. die vielen nützlichen Tools von Programmierern für Programmierer.
2. Die Entwickler machten sich auf Basis von fehlenden oder unzureichenden Informationen über die spätere Aufgabe eine eigene Vorstellung der Aufgabenlösung. Je nach dem, wie weit diese von der für die Aufgabe

typischen Bearbeitung abwich, entsteht so dann Schulungsbedarf bei der Benutzerseite. Diese werden dann nachträglich, zusätzlich zu ihrer fachlichen Qualifikation, auch noch im Umgang mit der Software im Sinne des Entwicklers geschult.

Der zweite Fall sollte im Sinne einer aufgabenangemessenen Gestaltung vermieden werden, denn es soll den späteren Benutzern ja möglichst die Last des Lernens wenigstens bezüglich des Mediums so weit wie möglich genommen werden.

Natürlich können neue Wege vorgesehen und eingesetzt werden, ggf. auch unter Zuhilfenahme von Schulungen und Lernphasen (ganz lassen sich diese derzeit noch nicht vermeiden). Man sollte sich dieses Mehraufwandes dann aber auch bewusst sein und ggf. auch diese Lernphasen im Umgang mit dem Medium berücksichtigen.

4.1.2 Selbstbeschreibungsfähigkeit

Definition

Selbstbeschreibungsfähigkeit soll den Benutzern den Einsatzzweck und den Funktionsumfang der Software jederzeit erkennbar machen. Daher sollten auf dem Bildschirm z.B. folgende Angaben verfügbar sein:

- Anwendungsgebiet,
- Anwendungsanleitung, auf Wunsch mit Beispielen,
- Funktionsangebot zur Aufgabenerledigung,
- Funktionsbeschreibungen,
- Handhabung des Dialogsystems,
- Voreinstellungen.

Jeder einzelne Dialogschritt sollte durch eine Rückmeldung in aufgabenangemessener Form unmittelbar verständlich sein. Um den Benutzern Dialogschritte verständlich zu machen, sollten bei der Gestaltung von Rückmeldungen und Erläuterungen folgende Dinge beachtet werden:

- Anpassung an die Kenntnisse der Benutzer,
- Verwendung einheitlicher und eindeutiger Begriffe, Zeichen, Symbole und Signale,
- Bezug zur aktuellen Anwendungssituation,
- Anzeige von Zustandsänderungen des Dialogsystems,
- Informationen über erforderliche Benutzeraktionen.

Beispiel Unangemessene Ausdrucksweise

„Suchbegriffe können wahlweise in Groß- oder Kleinschreibung eingegeben werden. Eine Endmaskierung („Trunkierung“ mit \$: kein, ein oder mehrere Zeichen dürfen folgen) ist möglich. Bsp.: Programmier\$“

(aus dem Bibliothekssystem der FU-Hagen; http://ub-www1.fernuni-hagen.de/scripts/fuh_opac.dll/opacmsk?benutzer=1; 1999)

Die verwendeten Formulierungen, wie zum Beispiel „Trunkierung“, sind wohl kaum aus dem Wortschatz des durchschnittlichen Studenten sondern eher aus der Beschreibung der Datenbank entnommen. Kaum ein Student wird etwas mit diesem Begriff anfangen können. Lediglich das Beispiel läßt erahnen, daß es um einen Platzhalter für beliebigen Text geht. Der Computerjargon wäre „Wildcard“ gewesen, wobei auch dies schon sehr benutzergruppenspezifisch gewählt wäre. Besser ist es in so einem Fall eine kurze Umschreibung in „natürlicher“ Sprache anzugeben (Lösungsvorschlag: „Sie können auch nach Teilen von Begriffen suchen. Geben sie dazu den zu suchenden Teil ein und markieren Sie das Ende mit dem \$-Zeichen“).

4.1.3 Steuerbarkeit

Definition

Steuerbarkeit liegt vor, wenn Benutzer nach Maßgabe der Arbeitsaufgabe sowohl Ablauf, Geschwindigkeit, Auswahl und Reihenfolge des Einsatzes von Arbeitsmitteln als auch Art und Umfang von Ein- und Ausgaben beeinflussen können. Dies heißt z.B., daß das Dialogsystem

- den Benutzer nicht zu einem bestimmten Arbeitsrhythmus zwingt, vor allem nicht durch Löschen von Bildschirmanzeigen;
- es ermöglicht, in leicht überschaubaren Dialogschritten vorzugehen und gegebenenfalls eine Zusammenfassung von einzelnen Dialogschritten vorzunehmen;
- eine Rücknahme mindestens des letzten Dialogschrittes möglich ist, sofern der ursprüngliche Anwendungszustand wiederherstellbar ist und die Arbeitsaufgaben es zulassen; zugleich müssen Systeme, die die Rücknahme von Dialogschritten teilweise erlauben, eine Warnung vor jedem nicht rücknehmbaren Schritt geben;
- der Benutzer bei erforderlichen Aktionen nach Bedarf zwischen Tastatur und anderen Eingabemitteln frei wählen kann;
- der Benutzer den Dialog unterbrechen oder beenden kann, soweit dies bei den vorgegebenen Arbeitsaufgaben möglich ist (wobei er entscheiden kann, ob der Dialog an der Unterbrechungsstelle fortgesetzt werden soll);

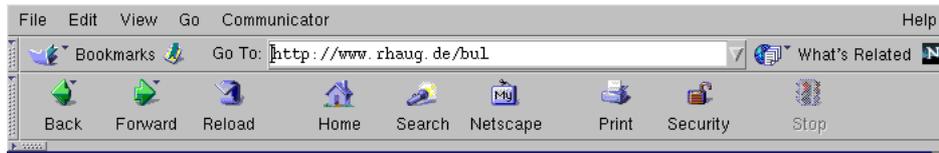


Abbildung 4.2: Gute Steuerbarkeit beim Web-Browsing (Netscape 4.6)

- der Benutzer zwischen Anwendungen wechseln kann, deren Benutzung zur Erfüllung seiner Aufgaben erforderlich ist, und Daten (z.B. Zeichen, Grafikelemente) zwischen diesen auf eine einfache Art und Weise auszutauschen kann;
- Benutzer auf zu erwartende erhebliche Abweichungen von den üblichen Antwortzeiten hingewiesen werden.

Zur Sicherung der Steuerbarkeit getroffene Maßnahmen dürfen die aufgabenbedingte Funktionserfüllung am Arbeitsplatz nicht beeinträchtigen.

Beispiel Gute Steuerbarkeit

Die in Abbildung 4.2 gezeigte Steuerleisten des Netscape Navigators erfüllen weitgehend die Anforderungen an die Steuerbarkeit einer Software:

- (+) Ein Abbruch des Ladens einer neuen Seite ist jederzeit möglich.
- (+) Die Software kann zu großen Teilen alternativ über Tastatur oder Maus bedient werden.
- (+) Der Benutzer gibt selbst das Tempo vor, in dem er auf die nächste Seite springt.
- (+) Es ist möglich, sich frei in den bereits vorher geladenen Seiten vor und zurück zu bewegen, so daß versehentliche Fehlnavigationen sofort ohne Aufwand ausgeglichen werden können.

4.1.4 Fehlertoleranz

Definition

Fehlertoleranz liegt vor, wenn der Benutzer das beabsichtigte Ergebnis trotz fehlerhafter Eingaben mit möglichst geringem Korrekturaufwand erreichen kann. Dies erfordert z.B., daß

- ein Dialogsystem sicherstellt, daß Eingaben der Benutzer nicht zu undefinierten Zuständen oder einem Systemzusammenbruch führen;
- Fehlermeldungen verständlich, sachlich, konstruktiv und einheitlich strukturiert formuliert sind;



Abbildung 4.3: Unkonstruktive Fehlermeldung

- Fehlermeldungen angemessene Informationen über Auftreten und Art des Fehlers enthalten und Korrekturmöglichkeiten aufzeigen;
- eine Prüfung, Veränderung und Kontrolle von Eingaben vor deren Ausführung möglich ist und Befehle mit großer Tragweite einer zusätzlichen Bestätigung bedürfen;
- der Benutzer bei automatischer Fehlerkorrektur durch das System über Korrekturmöglichkeiten sowie deren Ausführung informiert wird und diese beeinflussen kann;
- der Benutzer die Möglichkeit hat, Fehlerkorrekturen gegebenenfalls aufzuschieben.

Beispiel Mangelhafte Fehlermeldung

Wenn der Benutzer eine für das Programm unzulässige Eingabe gemacht also einen „Fehler“ begangen hat¹, dann reicht der alleinige Hinweis wie in Abbildung 4.3 dem Benutzer nicht aus, um ohne größeren Aufwand weiterarbeiten zu können. Im Zweifelsfall muß er nun suchen, auf welche seiner Eingaben sich diese Fehlermeldung bezieht und wie er es richtig machen könnte. Eine mögliche Lösung wäre zum Beispiel „Bitte geben Sie für den Tag eine Zahl zwischen 1 und 31 ein.“ und der Cursor wird anschließend in dem fehlerhaften Feld plaziert.

Noch schlimmer sind technische Fehlermeldungen, die direkt an den (nicht technisch versierten) Benutzer weitergereicht werden:

„Forms: FRM 40735: KEY HELP trigger raised unhandled exception VALUE_ERROR.“ (Aus einem Verwaltungsprogramm nach Drücken des Hilfe-Buttons)

4.1.5 Erwartungskonformität

Definition

Erwartungskonformität erfordert sowohl ein in sich konsistentes Dialogverhalten, als auch Übereinstimmung mit allgemeinen Konventionen und mit den

¹Prinzipiell ist es meist davon auszugehen, daß der Fehler bei der Maschine bzw. der eingesetzten Software liegt.

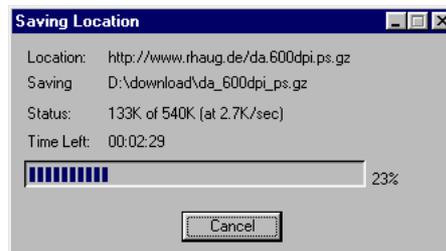


Abbildung 4.4: Erwartungskonformer Dialog, der alle notwendigen Eingriffsmöglichkeiten und Informationen bietet.

Kenntnissen und Erfahrungen, die Benutzer aus ihrem Arbeitsgebiet und ihrer Ausbildung mitbringen. Damit bildet sie eine Art Bindeglied zwischen den Forderungen 21.3 und 21.4 der Bildschirmarbeitsverordnung. Erwartungskonformität lässt sich z.B. erreichen, wenn

- Verhalten und Erscheinungsbild der an einem Arbeitsplatz eingesetzten Dialogsysteme möglichst einheitlich sind;
- ein den Benutzern bei der Erledigung ihrer Arbeitsaufgaben vertrauter Wortschatz verwendet wird;
- Aktionen der Benutzer eine unmittelbare Rückmeldung bewirken;
- Antwortzeiten der Aufgabenstellung entsprechen und allgemein akzeptabel sind,
- Leuchtdichte- und Farbcodierungen konsistent verwendet werden;
- Möglichkeiten zur Änderung des Dialogzustands einheitlich und ständig zur Verfügung stehen, damit innerhalb eines Arbeitsablaufs genau definierte und in ihrer Wirkung gleiche Zustände erreichbar sind;
- Benutzer bei ähnlichen Arbeitsaufgaben allgemein anwendbare Prozeduren zu deren Erledigung entwickeln und diese mit möglichst einheitlichen Dialogschritten und möglichst einheitlicher Wirkung bearbeiten können.

Beispiel Erwartungskonformer Dateiübertragungsdialog

Der in Abbildung 4.4 dargestellte Dialog erfüllt die Anforderungen an eine erwartungskonforme Gestaltung:

- Antwortzeiten auf Eingaben müssen kurz sein. Zumindest ein Echo und/oder eine Bestätigung der Eingabe müssen unmittelbar erfolgen!
- Während der Wartezeit sollte ein Abbruch der Aktion möglich sein.
- Wartezeiten werden als weniger störend empfunden, wenn der Benutzer über Ihren Grund und ihre Dauer informiert ist.

- Der Benutzer sollte erkennen können, ob das Programm sich noch in einem definierten Zustand befindet oder er einen neuen Versuch unternehmen soll.

4.1.6 Individualisierbarkeit

Definition

Individualisierbarkeit beinhaltet die Anpaßbarkeit eines Dialogsystems an die Erfordernisse der Arbeitsaufgabe sowie an individuelle Fähigkeiten und Vorlieben den Benutzer. Dies entbindet die Softwareentwickler allerdings nicht von ihrer Verantwortung für ergonomisches Design. Überdies sind Grenzen der Individualisierbarkeit dort zu setzen, wo Benutzer ansonsten sich selbst oder andere gefährden bzw. beeinträchtigen könnten (z.B. durch sinnesphysiologisch unangemessene Farbkombinationen, Lautstärken akustischer Rückmeldungen u.a.m.). Individualisierbarkeit liegt z.B. vor, wenn

- der Benutzer die Geschwindigkeit von Ein- und Ausgabefunktionen beeinflussen kann;
- dem Benutzer mehrere Dialogtechniken zur Verfügung stehen;
- der Benutzer zwischen alternativen Formen der Darstellung gemäß persönlichen Präferenzen oder der Komplexität der zu verarbeitenden Informationen wählen kann;
- das Dialogsystem an die Sprache, Fähigkeiten und Fertigkeiten eines Benutzers angepaßt und die Anpassung gespeichert und reaktiviert werden kann;
- der Benutzer den Detaillierungsgrad von Ausgaben (z.B. Fehlermeldungen, Hilfeinformationen) entsprechend seinem Kenntnisstand verändern kann;
- der Benutzer eigene Funktionen hinzuzufügen und Folgen von Funktionen zu Prozeduren zusammenfassen kann.

Beispiel Gute Individualisierbarkeit bei Netscape

Das Beispiel in Abbildung 4.5 zeigt eine positive Implementierung der Individualisierbarkeit. Zu bemängeln bleibt aber, daß das Programm dem Benutzer erlaubt, ungeeignete oder zu kleine Schriftarten einzustellen.

4.1.7 Lernförderlichkeit

Definition

Lernförderlichkeit liegt vor, wenn der Benutzer vom Dialogsystem selbst beim Erlernen von dessen Funktionsweise unterstützt und angeleitet wird. Dies kann z.B. erreicht werden, wenn

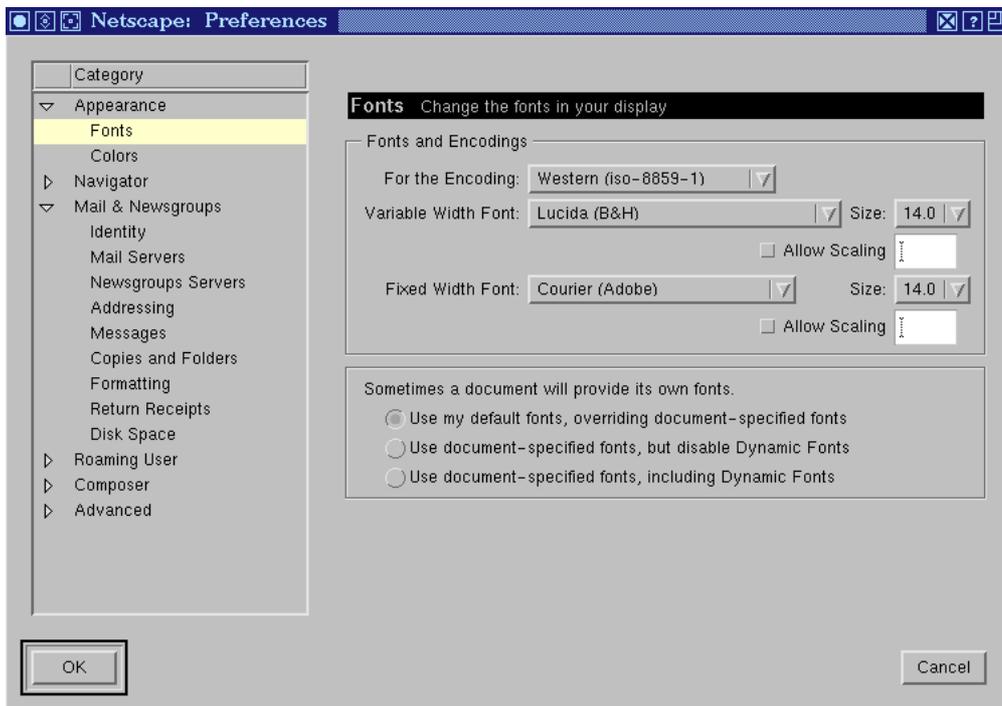


Abbildung 4.5: Der Benutzer kann hier zum Beispiel die verwendeten Schriftarten selbst wählen, um sie seinen Sehgewohnheiten anzupassen

4 Detaildarstellung grundlegender Regeln

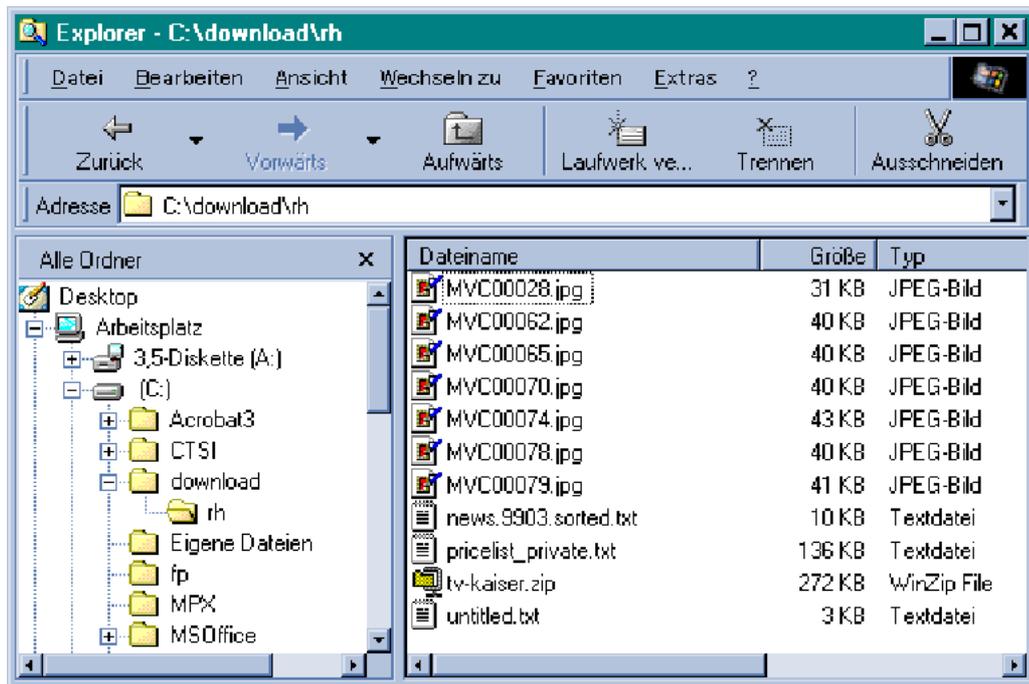


Abbildung 4.6: Lernförderliche Baumstruktur

- Konzepte und Regeln der Software so erläutert werden, daß deren Zweck, Aufbau, Möglichkeiten und Besonderheiten leicht behalten werden können;
- Lernstrategien wie exploratives Lernen, Lernen am Beispiel, verständnisorientiertes Lernen während der Arbeit mit dem System eingesetzt werden können.

Hinweis: Hier ist nicht die Förderung des Lernens *mit* der Software gemeint, sondern lediglich die Erlernung der Softwarebedienung an sich. Natürlich sollte Software im Rahmen der virtuellen Fachhochschule auch das Lernen an sich fördern. Dies wäre hier die zentrale Aufgabe und gehört somit bereits zur Aufgabenangemessenheit.

Beispiel Lernförderlichkeit durch grafische Darstellung

Die grafische Darstellung in Abbildung 4.6 erleichtert es dem Benutzer das Prinzip der Verzeichnisstruktur zu erlernen. Die zugeordneten Icons geben Hinweise auf den Inhalt der Dateien. Der Benutzer kann mit „Zurück“ und „Vorwärts“ durch seine eigenen letzten Schritte navigieren. Die Darstellung ist zusätzlich konsistent zur verbreiteten Bedienung von Browsern (vgl. Abb. 4.2 Seite 47). All dies nimmt dem Benutzer zum einen im Sinne der Selbstbeschreibungsfähigkeit viel Lernaufwand ab und zum anderen erleichtert es das Verstehen des Dateimodells.

4.2 Regeln zur Informationsdarstellung

Neben dem dynamischen Dialogablauf bestimmt die eigentliche Darstellung der Informationen wesentlich die Benutzbarkeit einer Software oder einer Web-Präsentation. Schließlich beruhen die Dialoge und die sinnvolle aktive Anwendung einer Software auf den Informationen, die der Benutzer von ihr erhält. Zunächst beschränkt sich diese Sammlung auf die Darstellung an Bildschirmen, doch sie ist auch auf andere Medien übertragbar (z.B. Ton).

4.2.1 Darstellungsprinzipien und Grundregeln

Die Begriffe stammen aus der ISO 9241 Teil 12, die sich mit der Informationsdarstellung befaßt. Daraus stammt auch diese folgende Prinzipienliste:

- **Klarheit**
Der Informationsinhalt wird schnell und genau vermittelt
- **Unterscheidbarkeit**
Die angezeigte Information kann genau unterschieden werden (z.B. als Überschrift)
- **Kürze**
Die Benutzer werden nicht mit übermäßigen Mengen von Information überlastet
- **Konsistenz**
Einheitliche Gestaltung und Bedienung, Erwartungskonformität (z.B. bei der Verwendung von Unterstreichungen)
- **Entdeckbarkeit**
Die Aufmerksamkeit des Benutzers wird zur benötigten Information gelenkt
- **Lesbarkeit**
Die Information ist leicht zu lesen.
- **Verständlichkeit**
Die Bedeutung ist leicht verständlich, eindeutig, interpretierbar und erkennbar

Um diese Prinzipien zu erfüllen, gibt es ganz allgemeine Gestaltungsregeln:

- **So wenig wie möglich und so viel wie nötig**
Eigentlich eine Selbstverständlichkeit. Informationen sollten wirklich nur dann für die permanente Darstellung vorgesehen werden, wenn sie tatsächlich auch die ganze Zeit benötigt werden.
- **Maximal 40% der Bildschirmfläche nutzen**
Freie Flächen erleichtern es dem Benutzer sich zu orientieren. Informationsgruppen und andere räumliche Hervorhebungen und Strukturierungen sind leichter und schneller wahrzunehmen. Insbesondere wenn

immer wieder ähnliche Informationen in kurzer Zeit gefunden werden sollen, ist dies leichter, wenn der Bildschirm klar gegliedert ist.

- **Die Funktion bestimmt die Form**

Dies bedeutet nicht, daß nun die Funktion allein die Form vorgibt, sondern daß sie Vorrang vor designerischen Aspekten haben soll. Dabei kann die Funktion einer Seite wiederum das Design in den Vordergrund stellen (Stichwort „Werbung“). In der Regel aber sollte zunächst ein funktionaler Entwurf erstellt werden, der dann designerisch nachbearbeitet wird.

- **Weniger ist mehr**

Ein sparsamer Umgang mit Attributen (unterstreichen, blinken, Farbe etc.) ist immer anzuraten. Der inflationäre Gebrauch solcher Hervorhebungen führt dazu, daß zum einen die Hervorhebungen ihren Sinn nicht mehr erfüllen, da sie in all den anderen Hervorhebungen untergehen, und zum anderen die Fülle der Kodierungen die Übersichtlichkeit und das Design ruinieren.

In den folgenden Kapiteln werden die einzelnen Prinzipien näher erläutert und mit Beispielen versehen. Genauere Vorgaben und Regeln sind nicht sinnvoll, da sie den Designspielraum z.T. unzulässig einschränken würden (siehe allerdings [Ballstaedt 97]). Es hängt ganz von der betrachteten Zielgruppe und Funktion der Software ab, wie eine Seite letztendlich gestaltet wird.

Die Darstellungsprinzipien sind nicht immer scharf voneinander abzugrenzen und können ineinander übergehen. So ist zum Beispiel die Forderung nach Klarheit kaum von den anderen Prinzipien zu trennen.

Beispiel

Im abgebildeten Beispiel (Abbildung 4.7) sind alle genannten Darstellungsprinzipien mehr oder weniger stark verletzt worden.

Weder ist schnell und genau zu erkennen, was der Autor mit dieser Seite bezwecken möchte (Klarheit), sind Texte, Bilder und Hintergrund unterscheidbar, noch ist diese Seite konsistent zu einem bekannten Schema oder auch nur zu den Folgeseiten. Die Farbwahl (dunkelblaue Links auf schwarzem Grund) beeinträchtigt die Entdeckbarkeit („wo stehen denn noch Texte?“) und vor allem die Lesbarkeit. Letzten Endes ist die Verständlichkeit der verwendeten Ausdrucksweise („manueller Counter“) bzw. der eingesetzten Begriffe („Zu Alex“) für den neuen außenstehenden Benutzer völlig unklar.

Diese Mängel sind vor dem Hintergrund der vorgesehenen Benutzung zu bewerten. Die im Beispiel gezeigte Seite war als Werbung für einen Web-Dienstleister gedacht, der damit seine Dienste als *Designer* anbot. Dieser Zweck wird durch die Gestaltung der Seite völlig konterkariert.

4.2 Regeln zur Informationsdarstellung



Abbildung 4.7: Beispiel einer Webseite, die eigentlich alle Darstellungsprinzipien zugleich verletzt

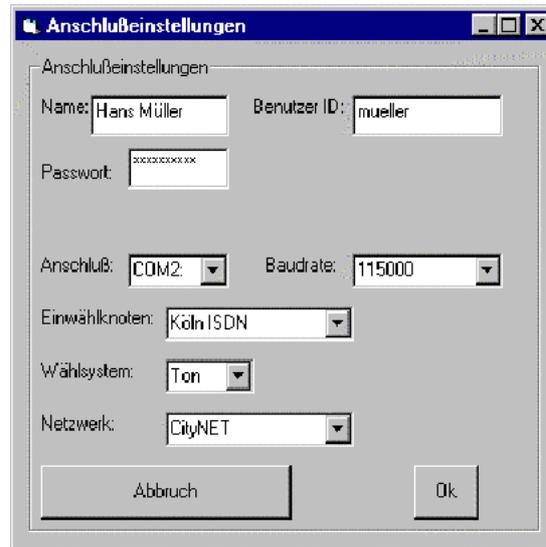


Abbildung 4.8: Unklar gestalteter Dialog

4.2.2 Klarheit

Definition

Das Ziel, Informationen **schnell** und **genau** zu vermitteln, sollte eigentlich jede Seite haben, die nicht als Rätsel gedacht ist. Dem Benutzer wird so erspart, daß er zunächst einen Teil seiner (mentalen) Leistungsfähigkeit darauf verwenden muß, überhaupt die zentralen Informationen einer Darstellung zu erfassen.

Klarheit wird durch eine gut strukturierte Darstellung erreicht, die schon durch die räumliche Anordnung (aber auch durch den Einsatz von weiteren Kodierungstechniken) den Benutzer bei der Benutzung der Seite unterstützt.

Beispiel Unklare Strukturierung

Das Beispiel (Abbildung 4.8) zeigt einen unklar gestalteten Dialog² :

- Die Informationen sind scheinbar wahllos auf dem Bildschirm verteilt und haben keine einheitlichen **Fluchtlinien**.
- Die **Gruppierung** mittels eines Rahmens ist überflüssig, wenn darin der gesamte Inhalt enthalten ist.
- Die **unterschiedliche Größe** der Schaltflächen suggeriert eine (falsche) zu bevorzugende Auswahl (nämlich „Abbruch“).

²Der Vollständigkeit halber sei erwähnt, daß auch die angebotenen Optionen „Minimieren“ und „Maximieren“ (Icons in der rechten oberen Ecke des Dialoges) hier falsch sind, da sie nicht sinnvoll benutzt werden können.



Abbildung 4.9: Aufgeteilter und verbesserter Dialog (I)

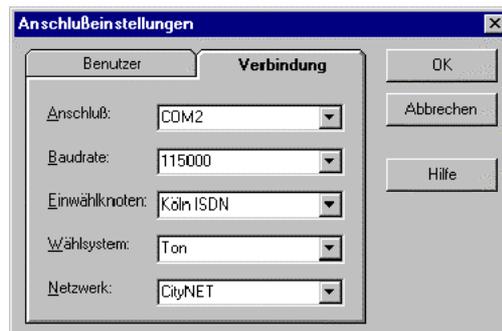


Abbildung 4.10: Aufgeteilter und verbesserter Dialog (II)

Durch einige wenige Veränderungen lassen sich die genannten Mängel abstellen (siehe Abbildung 4.9 und 4.10). Der Dialog wurde nach Inhalten getrennt auf zwei kleinere Einheiten (hier Karteikarten) verteilt und die Fluchtlinien wurden angepaßt. Die Schaltflächen wurden konsistent zu anderen Anwendungen und gleich groß gestaltet.

4.2.3 Unterscheidbarkeit

Definition

Unterscheidbarkeit spielt in zweierlei Hinsicht eine Rolle: Zum Einen bzgl. der Unterscheidbarkeit von Hintergrund und Vordergrund, die unter dem Begriff „Lesbarkeit“ im Abschnitt 4.2.7 noch genauer behandelt wird. Unterscheidbarkeit betrifft aber auch die Unterscheidung von verschiedenartigen Objekten. Sehen sich Objekte zu ähnlich so werden sie vom Benutzer nicht oder nur mit sehr hohem Aufwand noch unterschieden. Der Benutzer wird dann vielleicht feststellen, daß Aktionen, die eben noch möglich waren, hier nun ausgeschlossen sind. Beispielsweise ist auf manchen Seiten nicht mehr zu unterscheiden, welche Bereiche aktivierbar sind (zum Beispiel Links) und welche rein informativen Charakter haben. Dies führt dazu, daß eigentlich aktivierbare Objekte nicht mehr als solche wahrgenommen werden und/oder daß der Benutzer frustriert versucht, eigentlich inaktive Objekte anzuklicken.

Beispiel Unterscheidung von Links

Ein Beispiel ist die Unterscheidung von bereits besuchten und noch neuen Links. Häufig werden die Farbeinstellungen so manipuliert, daß solche Links entweder gar nicht oder nur noch sehr schwer (zum Beispiel durch sehr ähnliche Farbtöne oder Helligkeiten) zu unterscheiden sind. Damit wird die intendierte Unterstützung der Benutzer bei der Navigation durch ein Dokument („Diesen Link habe ich schon von irgendwoher besucht.“) zunichte gemacht.

4.2.4 Kürze

Definition

Auch überflüssige Informationen belasten den Benutzer. Ein gewisses Maß an Zusatzinformationen kann gerechtfertigt sein, viele Informationen sind aber nicht notwendig oder sinnvoll für die eigentliche Aufgabe. Diese sollten dann auch weggelassen werden.

Beispiel Überladene Web-Seite

Die in Abbildung 4.11 dargestellte Webseite soll dem Benutzer eine komfortable Oberfläche zum Lesen und Schreiben von Usenet-Nachrichten bieten.

4.2 Regeln zur Informationsdarstellung

The screenshot shows a Netscape Communicator browser window displaying the website www.deja.com. The browser's address bar shows the URL: `http://www.deja.com/[ST_rn=ps]/qs.xp?0P=dnquery.xp&ST=QS&QRY=de.alt.sci.ergonomie`. The website header features the 'deja.com' logo and a navigation menu with links for 'Home', 'Computing & Tech', and 'Internet Services'. Below the header, there is a search bar and a section titled 'Rate it!' with a rating scale for 'Concentric'. The main content area displays 'Discussion Search Results: "de.alt.sci.ergonomie"' and lists 'Top Forums related to de.alt.sci.ergonomie' and 'Messages related to de.alt.sci.ergonomie'. The messages are presented in a table with columns for Date, Subject, Forum, and Author.

| Date | Subject | Forum | Author |
|----------|--------------------------------|-----------------------|-----------------|
| 07/19/99 | <1999-05-25> Newsgroup-Ueber | de.comm.infosystems.w | Chris Kaschig - |
| 07/19/99 | Re: Help! | de.alt.sci.ergonomie | Ronald Hartwig |
| 07/16/99 | Help! | de.alt.sci.ergonomie | mark_schurz |
| 07/13/99 | cmsq cancel <7mdld2f\$aa0f20@f | de.alt.sci.ergonomie | W.KOK |
| 07/12/99 | Re: Flussdiagramme, Schaltkr | de.alt.sci.architektu | Jens Hoffmann |
| 07/08/99 | essai | de.alt.sci.ergonomie | Thibault |
| 07/12/99 | Re: Flussdiagramme, Schaltkr | de.alt.sci.architektu | Marcel Sauder |
| 07/12/99 | www.Software-Forum.de | de.alt.comm.isdn4linu | Christian Henke |
| 07/11/99 | AutoMoose NoCeM Spam Report | alt.nocem.misc | Cancelmoose[tm] |
| 07/09/99 | Re: Banner - nachweisbare Ef | de.alt.sci.ergonomie | Manuel Friedric |
| 07/09/99 | Fahrrad Werbung | de.alt.sci.ergonomie | Jens Wiese |

Abbildung 4.11: Überladene Web-Seite

Da der Betreiber aber auch versucht, ein Maximum an Werbung unterzubringen, kommt es zu einem Interessenkonflikt. Der Platz, der durch reine Werbezusätze zum Beispiel an den seitlichen Rändern verlorengeht, fehlt bei der Darstellung der Artikelüberschriften, die deshalb nur noch verkürzt dargestellt werden können. Letzlich sind aber gerade die Artikelüberschriften der von der Aufgabe her zentrale Bestandteil dieser Dienstleistung.

4.2.5 Konsistenz

Definition

Eine durchgängige Gestaltung erleichtert es dem Benutzer, sich in einer Umgebung zurechtzufinden. Wir alle kennen diese Konsistenz (-probleme), wenn wir in anderes Auto steigen. Wir sind bestimmte Anordnungen gewöhnt und müssen diese deshalb nicht bei jedem PKW neu erlernen (Beispiel: Anordnung der Pedale, Lenkrad und Instrumente etc.). Natürlich sind aber auch individuelle Unterscheidungen zulässig und erwünscht (zum Beispiel bei der designerischen Gestaltung des Armaturenbrettes). Bei jeder Designentscheidung sollte zunächst geprüft werden, ob es bereits eine existierende Entsprechung gibt und wie verbreitet diese bei der betrachteten Benutzergruppe sein wird (Konsistenz zu bestehenden Fremdprodukten). Außerdem ist innerhalb eines Produktes (Software oder Web-Seite) eine durchgängige Bedienung und Gestaltung vorzusehen.

Soll nun von einer solchen bestehenden Konvention abgewichen werden, so sind die Folgen dieser Abweichung abzuschätzen. Dabei sind die folgenden Faktoren zu berücksichtigen und abzuwägen:

- Bringt diese Abweichung einen Mehrwert? Dieser kann in einer besseren Bedienbarkeit, aber auch in einem gefälligeren Design bestehen.
- Wie bekannt und verbreitet ist die Alternative, von der abgewichen wird? Ist diese nur in speziellen Produkten realisiert, die die späteren Benutzer voraussichtlich nicht wesentlich häufiger benutzen werden als das eigene, so kann eine Abweichung tolerierbar sein.
- Verletzt die neue Alternative bestehende gestalterische Regeln? Häufig sind Konventionen über Jahre entwickelt und haben sich als optimal herausgestellt. Bestehende Alternativen wurden vielleicht bereits früher aus bestimmten Gründen abgelehnt.

4.2.6 Entdeckbarkeit

Definition

Die Aufmerksamkeit des Betrachters kann durch Gestaltung gesteuert werden. Dabei ist zu beachten, daß auch die wirklich bedeutsamen und für die Aufgabe notwendigen Informationen in den Vordergrund gestellt werden. Durch Überladung oder ungeschickte Farbgestaltung (siehe Abbildung 4.7)

kann es dazu kommen, daß der Benutzer wichtige Teile schlichtweg nicht entdeckt. Wie bereits dargestellt, ist auf eine übersichtliche Strukturierung und gute Unterscheidbarkeit von Informationen zu achten.

4.2.7 Lesbarkeit

Die Lesbarkeit betrifft hauptsächlich die elementare Forderung nach einer Lesbarkeit der Schrift einer Software oder Webseite. Durch falsche oder ungünstige Vorgaben können Schriften bis zur Unleserlichkeit verkleinert werden. Auch die Benutzung unzulässiger Farbkombinationen kann die Lesbarkeit drastisch senken. Als Faustregel gelten die folgenden Leitsätze:

- Die Schrift sollte auf einem defaultmäßig eingestellten Rechner mit einer für den Monitor optimalen Bildschirmauflösung (also 800x600 bei 15“ und 1024x768 bei 17“) noch mindestens 3mm groß sein.
- Aufgrund der größeren Auflösung eines Monitors (verglichen mit dem Papierpendant), sind auch im Fließtext serifenlose Schriften (zum Beispiel Arial, Helvetica, Lucida) den sonst üblichen (und teilweise auch per default eingestellten) Buch- und Zeitungsschriften (mit Serifen) vorzuziehen.
- „Bleiwüsten“ und überlange Zeilen sollten vermieden werden. Das Auge benötigt Strukturierungshilfen (insbesondere aufgrund der Oberlängen einer Schrift), um sich immer wieder zurecht zu finden und Anschluß an das letzte gelesene Wort zu halten. Bei zu langen Zeilen (> 60 Zeichen/Zeile) besteht die Gefahr, daß man beim Zeilensprung Schwierigkeiten bekommt.
- Die durchgehende Verwendung von Großbuchstaben sollte vermieden werden (siehe Beispiel).

Besonders bei längeren Textpassagen sind diese Regeln elementar wichtig, da sonst eine Ermüdung des Betrachters eintritt, die die Leistungsfähigkeit erheblich einschränkt.

Beispiel Großbuchstaben

Die durchgehende Verwendung von Großbuchstaben (siehe Abb 4.12) erschwert dem Leser die gewohnte Leseweise. Bereits bei diesen kurzen Feldbezeichnern wird der Benutzer erheblich beeinträchtigt. Schließlich sind wir es gewöhnt Kleinschrift zu lesen und die Großbuchstaben als besondere Markierung von bestimmten Wortarten (z.B. Substantiven) oder Positionen (Satzanfang) wahrzunehmen (siehe Abbildung 4.13). Wird entgegen dieser Gewöhnung gestaltet, so beginnen wir die Wörter mehr oder weniger buchstabenweise zusammenzusetzen, was zu einem erheblichen Mehraufwand führt.

4 Detaildarstellung grundlegender Regeln

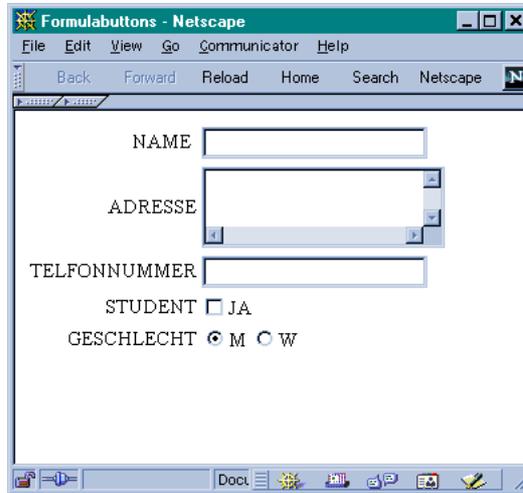


Abbildung 4.12: Schlechte Lesbarkeit wegen durchgehender Großschreibung

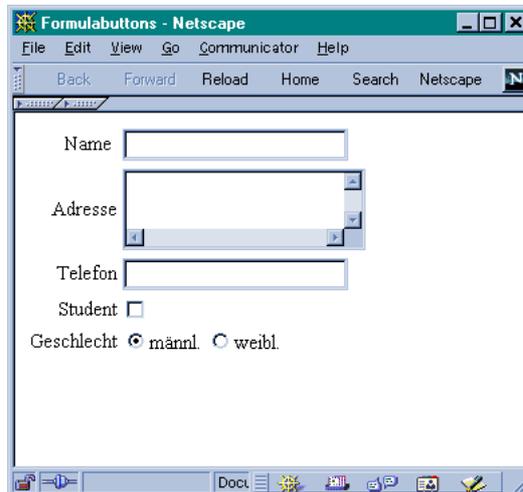


Abbildung 4.13: Verbesserte Lesbarkeit durch normale Groß-/Klein-Schreibung



Abbildung 4.14: Unklare Icons (Mac-Applikationen)

4.2.8 Verständlichkeit

Definition

Zum Einen ist bereits im Rahmen der Dialoggestaltung darauf hingewiesen worden, daß die Ausdrucksweise der Benutzergruppe angepaßt sein muß. Darüber hinaus sind Informationen insgesamt „verständlich“ darzustellen. Dies betrifft die Struktur einer Seite („Warum gehören diese Elemente zusammen?“, „Was haben diese Elemente für Gemeinsamkeiten?“) und auch die Kodierung von Informationen, zum Beispiel zu Symbolen. Alles sollte so gestaltet sein, daß der Benutzer ohne erheblichen (idealerweise ohne jeglichen) zusätzlichen mentalen Aufwand die Gestaltung versteht und sinnvoll nutzen kann.

Beispiel Unverständliche Icons

Die in Abb. 4.14 gezeigten Icons sind auf den ersten Blick nicht alle selbsterklärend oder unmittelbar verständlich. Dabei bestehen folgende positive wie negative Möglichkeiten:

- (-) Manche Symbole sind völlig unabhängig von der dahinterliegenden Funktion bzw. Objekteigenschaft (Bsp.: „BBEdit Lite 4.0“) und orientieren sich stattdessen am Firmendesign des Herstellers. Wenn dies allerdings verbreitet genug ist und für das Programm als bekannt vorausgesetzt werden kann (Bsp. „Microsoft Excel“), kann es in Einzelfällen hingenommen werden (Immerhin liegt hinter dem „X“ des Excel-Logos ja auch noch ein Tabellenblatt).
- (-) Symbole zeigen nur einen Teilbereich oder gar nicht vorhandene Funktionalitäten. Beispielsweise wird im gezeigten Screenshot der „Netscape Communicator“ als Texteditor symbolisiert, was er aber nur im speziellen Fall als Mail-Client und dann auch nur bedingt ist.
- (-) Unpraktisch sind auch Symbole, die zu klein sind, um sie noch sinnvoll zu erkennen (Bsp. „Drop PS“).
- (-) Zu erheblicher Verwechslungsgefahr führt schließlich die Verwendung von ähnlichen Symbolen bei grundverschiedener Funktion (hier beispielsweise „Stuffit Expander“ und „NCSA Telnet“).

4 Detaildarstellung grundlegender Regeln

- (+) Sinnvoll ist die Benutzung von naheliegenden Symbolen, wie zum Beispiel eines „Bildes“ als Symbol für ein „Bild“-Erstellungsprogramm wie „Adobe Illustrator“.

5 Entwurfscheckliste

Die folgenden Anforderungen sind aus den praktischen Erfahrungen bei der Evaluation von ersten Prototypen heraus entstanden und bieten einen ersten Anhaltspunkt, was auf jeden Fall zu vermeiden bzw. zu machen ist. Einen vollständigeren Zusammenstellung von Anforderungen steht mittlerweile in einem eigenständigen Dokument zur Verfügung (siehe [Hartwig, Triebe & Herczeg 2002]).

5.1 Schrift

Anforderung 1 (Schriftgröße)

| |
|-----------------|
| Mindestens 10pt |
|-----------------|

Die Schrifthöhe sollte auf einem Monitor bei einer Auflösung von 800x600 mindestens 3mm betragen, wenn man von einem Betrachtungsabstand von 50cm ausgeht. In der Praxis wird dies durch Schriftarten mit mindestens 10pt erreicht.

Hinweis: Da Apple-Rechner eine geringere Bildschirmauflösung haben (75dpi statt 96dpi) werden Schriften dort kleiner dargestellt. Die genannte Regel gilt so zunächst nur für Windows-Rechner.

Anforderung 2 (Schriftart)

| |
|-----------------------|
| Serifenlose Schriften |
|-----------------------|

Da auf dem Bildschirm eine wesentliche schlechtere Auflösung zur Verfügung steht als auf einem Blatt Papier (ca. 96dpi statt 600dpi o. 1200dpi), neigen Schriften mit kleinen waagerechten Strichen am Ende des Buchstabens (Serifen) zum „verkleben“, d.h. die Buchstaben wirken fälschlicherweise miteinander verbunden. Dies senkt die Lesbarkeit erheblich¹.

Bekannte Serifenlose Schriften:

- Arial, Verdana, Tahoma (Windows)
- Helvetica (Linux, Solaris)

Nicht gut geeignet ist hingegen die Serifenschrift „Times New Roman“, die aber leider bei vielen Browsern als Default benutzt wird.

¹Diese Serifen wurden zur Verbesserung des Leseflusses eingeführt, um dem Auge einen besseren Zeilenhalt zu geben. Ironischerweise verkehrt sich dieser Effekt am Bildschirm bei der derzeitigen Technologie noch ins Gegenteil.

Anforderung 3 (Zeilenlänge)

Nicht mehr als ca. 60 Zeichen pro Zeile

Im Buchdruck gibt es eine Regel, dass Zeilen nicht länger als das doppelte Alphabet werden sollten, d.h. ca. $30 \cdot 2 = 60$ Zeichen pro Zeile. Längere Zeilen führen zu Problemen beim Zeilensprung, so dass das Auge in der Zeile sich neu orientieren muss. Zu kurze Zeilen hingegen erzeugen durch den notwendigen Umbruch störende Leerstellen.

5.2 Ausdruck

Anforderung 4 (Höflichkeit)

„Du“ vermeiden

Bei den potentiellen Benutzern ist nicht davon auszugehen, dass diese vorbehaltlos geduzt wollen werden. Die Ansprache der Benutzer sollte dementsprechend neutral sein oder ggf. auf „Sie“ basieren.

Anforderung 5 (Verständlichkeit)

Technikjargon vermeiden

Abgesehen von explizit dieses Thema behandelnden Kursen und Modulen, sind technische Fachausdrücke wie „Splash-Screen“ oder auch nur „Icon“ und „URL“ entweder zu vermeiden, oder wenn notwendig, zu erläutern. Auch techniknahes Deutschenglisch wie „Verlinken“, „Downloaden“ etc. sind nicht unproblematisch.

Anforderung 6 (Freundlichkeit)

Nicht den Benutzer „anschreien“

Vermeiden Sie Ausdrücke wie „FALSCH!“ oder „Benutzerfehler“. Die Großschreibung wird im Internet als „Anschreien“ gedeutet und ist auch ansonsten nicht sehr freundlich. Stattdessen sollten mildere Ausdrucksweisen und konstruktive Rückmeldungen gewählt werden: „Bitte geben Sie eine Zahl zwischen 1 und 31 ein.“ statt „Falsche Eingabe! Data out of range“.

5.3 Gestaltung

Anforderung 7 (Gesetz der Nähe)

Was zusammengehört sollte auch zusammenstehen, aber was nicht dazu gehört, sollte räumlich auch getrennt sein.

Inhalte, die nahe beieinander stehen, werden von uns automatisch zunächst auch in einen inhaltlichen Zusammenhang gebracht. Daraus folgt, dass zum Beispiel Icons, die einen Link beschreiben, auch in dessen Nähe angebracht sein müssen und nicht davon vertikal und horizontal versetzt. Umgekehrt sollten Fehlinterpretationen vermieden werden, also sollten zum Beispiel Schriften und Bilder nicht zusammenstehen, wenn sie nichts miteinander zu tun haben.

Auch die Gliederung von Tabellen sollte die inhaltlichen Aspekte widerspiegeln und sich nicht primär am Platzangebot orientieren.

Anforderung 8 (Ähnlichkeit konsistent nutzen)

Ähnliche Objekte sollten auch ähnliche Aktionen auslösen.

Bei ähnlich aussehenden Elementen (zum Beispiel Icons oder Buttons) erwartet der Benutzer auch ähnliche dahinter verborgene Funktionalitäten. Umgekehrt erwarten Benutzer auch aus anderen Anwendungen bekannte Icons wiederzuerkennen. Dies gilt auch für ähnlich klingende Worte und Bezeichner.

Daraus folgt zum Beispiel auch, dass Verweise (Links) dem im Web verwendeten Standard gemäß unterstrichen werden müssen und die Farbkodierung sich an die ebenfalls defaultmäßig existierenden Konventionen (dunkelblau und lila) halten sollten. Von der Farbkodierung kann im Einzelfall aus gestalterischen Gründen abgewichen werden, von der Kodierung „Unterstreichung“ hingegen nicht.

Anforderung 9 (Falsche Ähnlichkeit vermeiden)

Ähnlichkeiten sollten bei Verwechslungsgefahr vermieden werden.

Sind im Gegensatz zum vorherigen Punkt verschiedene Funktionalitäten gemeint, so sollten sich auch die damit verknüpften Elemente deutlich unterscheiden.

Auch sollten sehr unklare Ähnlichkeiten vermieden werden, bei denen der Benutzer zunächst raten müsste, was ein Symbol oder ein Element für eine Bedeutung haben könnte. Stattdessen ist dann eine textbasierte Form vorzuziehen.

Anforderung 10 (Hintergrund)

Muster und gesättigte Farben vermeiden

Wenn Hintergründe gefärbt sein sollen, so ist darauf zu achten, dass zum einen ein ausreichender Kontrast zur Schrift erhalten bleibt, bei schwarzer Schrift also der Hintergrund nicht zu dunkel wird, zum anderen dass die Farben nicht zu aufdringlich, also gesättigt sind. Stattdessen sind Pastellvarianten zu verwenden.

Strukturierte Hintergründe sind in der Regel ganz zu vermeiden, da sie die ohnehin nicht optimale Lesbarkeit von Schriften auf Computerbildschirmen noch weiter vermindern und das Lesen unnötig erschweren

Anforderung 11 (Ablenkung)

Ungewollte und/oder dauerhafte Ablenkungen vermeiden.

Die Wahrnehmung des Menschen ist darauf spezialisiert, auf Bewegungen am Rande des Sichtfeldes zu achten und ggf. zu reagieren. Dies führt dazu, dass man als Mensch förmlich gezwungen ist, auf Animationen und andere bewegte Elemente zu schauen, auch wenn diese bereits als nicht benötigt identifiziert wurden. Dies führt zu einer erheblichen Ablenkung und Reduzierung der Leistungsfähigkeit und sollte deshalb vermieden werden. Bewegte Inhalte sollten erst auf Benutzerwunsch hin aktiv werden oder zumindest durch Benutzereingriff gestoppt werden können. Animierte Grafiken als Logo auf jeder Seite sind prinzipiell abzulehnen.

Anforderung 12 (Anordnung)

Die Anordnung der Inhalte sollte dem Nutzungsablauf angemessen sein

Die Leserichtung ist von links nach rechts und von oben nach unten. Werden nun auf einer Seite verschiedene Elemente angeordnet, bei denen die Reihenfolge der Anwendung klar ist (z.B. erst Text lesen, dann Kommentar schreiben und dann „Speichern“-Button drücken), so sollten diese ebenfalls in dieser Reihenfolge angeordnet sein.

5.4 Benutzerunterstützung

Anforderung 13 (Unterbrechungsfreies und effizientes Arbeiten)

Zusätzliche Arbeitsschritte sollten immer durch das Konzept zu rechtfertigen sein und für den Benutzer einen Gewinn bringen. Sie sollten möglichst außerhalb des eigentlichen Hauptarbeitsablaufes gebündelt werden.

Prinzipiell ist ein Studienmodul an den Bedürfnissen der Benutzer und nicht der Autoren auszurichten (siehe auch nächsten Punkt). Wenn aber zusätzlicher Aufwand notwendig ist, zum Beispiel das Installieren eines Plug-Ins, so sollte dies im Rahmen eines besonderen Arbeitsschrittes und nicht erst während der Lernphase stattfinden. So sollten benötigte Plug-Ins bereits auf einer einleitenden Seite eingeführt und überprüft werden. Auch sollte soweit möglich die Notwendigkeit zusätzlichen Aufwandes begründet werden.

Wichtig ist auch die rechtzeitige Information des Benutzers, d.h. wenn alternative oder zusätzliche Möglichkeiten zur Bearbeitung eines Lernmoduls bestehen, sollten diese bereits am Beginn des Moduls erläutert werden.

Anforderung 14 (Unterbrechbarkeit)

Der Benutzer sollte seine Tätigkeit selbst unterbrechen und ohne übermäßigen Aufwand später am selben Punkt wieder aufnehmen können.

Im Einsatz von Lernmodulen unter Alltagsbedingungen sind auch Störungen und Unterbrechungen denkbar und wahrscheinlich. Diese können zum Beispiel organisatorisch (Pausenregelungen) oder sozial (Besuch, Telefonanrufe) bedingt sein. Es ist dann notwendig, dass die Benutzer ihre Arbeit ohne erheblichen Datenverlust später dort fortsetzen können.

Praktisch heißt dies, dass Bookmarks oder andere Hilfen vorzusehen sind, um den Wiedereinstiegspunkt zu markieren. Ausserdem sollten angefangene Arbeiten nicht verloren gehen. Bei sehr kleinen Aufgaben (z.B. ein oder zwei Fragen) kann davon allerdings abgesehen werden.

Anforderung 15 (Gewohnte Arbeitsweisen unterstützen)

Frames vermeiden, Standard-Browserfunktionen anbieten

Auch wenn Lösungen wie „Frames“ den Implementierungsaufwand senken, so bedeuten sie für die Benutzer erhebliche Nachteile, denn Standardvorgehensweisen, wie das Ausdrucken einer Seite, Setzen eines Bookmarks oder das Versenden per E-Mail sind dann nur noch (über den meisten Benutzern unbekannte) Umwege möglich. Es ist deshalb nicht ratsam, Frames oder (Zusatz-)Fenster ohne Symbol- und Menüleisten einzusetzen. Ausserdem sollten die Titel der einzelnen Seiten so sprechend sein, dass sie in einer eigenen Bookmark-Sammlung auch noch hilfreich sind.

Zur gewohnten Arbeitsweise wird in den meisten Modulen auch das lineare Durchgehen der Einheiten gehören, so dass eine Navigation „Vorherige Seite“ und „Nächste Seite“ gut sichtbar immer verfügbar sein sollte. Die Begriffe „Zurück“, „Vor“ und „Vorwärts“ sollten nicht ohne Erläuterung zur Navigation verwendet werden, da sie bereits in den Browsern für das Navigieren innerhalb der History benötigt werden.

Anforderung 16 (Ungewohnte Bedienung vermeiden)

| |
|---|
| Spezielle moduleigene Vorgehensweisen vermeiden |
|---|

Sofern nicht aus inhaltlicher Sicht vertretbar, sollten für das Studienmodul spezifische aber ansonsten unübliche Bedienkonzepte vermieden werden. Beispielsweise ist es nicht sinnvoll, eigene Eingabeformate für Suchfunktionen zu nutzen (oder Datenbankvorgaben wie das „%-Zeichen zu übernehmen) oder Drag & Drop unkritisch ohne weitere Erläuterung einzusetzen. Es sollte bedacht werden, dass jeder zusätzliche Bedienungsaufwand die verfügbare Lernkapazität für den eigentlichen Inhalt schmälert.

5.5 Information des Benutzers

Anforderung 17 (Klarheit der Handlungsmöglichkeiten)

| |
|--|
| Dem Benutzer sollte zu jeder Zeit bewußt sein, welche Handlungsmöglichkeiten ihm zur Verfügung stehen. |
|--|

Von didaktisch zu begründenden Ausnahmen abgesehen, sollte dem Benutzer durch die Anordnung und Gestaltung der Bedienelemente und Verweise zu jeder Zeit klar sein, welche Handlungsmöglichkeiten ihm gerade zur Verfügung stehen.

Dazu gehört zum Beispiel, dass in Grafiken eingebundene Links besonders gekennzeichnet werden. Bei Texten durch Kodierung als Links (Unterstreichung), in Grafiken („Clickable-Maps“) durch besondere Hinweise („Wählen Sie bitte das Telefon oder das Faxgerät aus.“) oder Gestaltung als übliches aktives Element (Button, Schieberegler, Eingabefeld etc.).

Ein wichtiger Punkt ist auch die Information über absichtlich **nicht** verfügbaren Optionen. Wenn eine Handlungsmöglichkeit ausnahmsweise nicht wie sonst üblich zur Verfügung steht, so ist diese nicht ersatzlos zu streichen sondern als nicht verfügbar („grau“) an gewohnter Stelle darzustellen, um so dem Benutzer anzuzeigen, dass es eine bewußte Entscheidung und nicht ein Implementierungsfehler ist. Ein Beispiel dafür wäre ein gesperrtes „Weiter“ zur nächsten Seite auf der letzten Seite, damit das Ende des Moduls als solches klar erkennbar ist.

Anforderung 18 (Statusanzeigen)

| |
|---|
| Wartezeiten müssen nachvollziehbar und abschätzbar sein |
|---|

Wenn Teile eines Moduls längere Ladezeiten benötigen, so sollte der Benutzer vorher darüber informiert werden („Diese Datei ist 1,8 MB groß und wird per ISDN ca. 2-10 Minuten benötigen.“), währenddessen die Möglichkeit zum Abbruch haben, wissen, wie lange es nun aktuell noch dauern wird und ob der Download noch aktiv ist.

Wartezeiten werden bereits ab 0,3 Sekunden als störend empfunden. Bei mehr als 3 Sekunden Wartezeit ohne Rückmeldung sind Benutzer bereits verunsichert und Wartezeiten über eine Minute werden ohne entsprechende Rückmeldungen als Absturz identifiziert.²

Anforderung 19 (Positionsanzeige)

Der Benutzer sollte wissen, wo er sich gerade befindet, möglichst auch relativ zum Gesamtinhalt.

Um dem Benutzer eine Einteilung seiner Zeit zu erleichtern, müssen ihm Hilfen zur Abschätzung der verbliebenen Inhalte gegeben werden, zum Beispiel durch eine Fortschrittsanzeige (Beispiel „Seite 5 von 55“). Ausserdem sollte immer klar sein, wo er sich im Kontext befindet (z.B. in einer Hierarchie).

Dabei ist zu beachten, dass diese Darstellung auch wirklich konsistent zur tatsächlichen Position ist. Wird die Position zum Beispiel in einem extra Fenster oder Frame dargestellt, so ist sicherzustellen, dass bei einem Wechsel der Seite, auch die Positionsdarstellung in dem zusätzlichen Fenster (sofern vorhanden) automatisch aktualisiert wird. Falls dies technisch nicht zuverlässig möglich ist, so sollte man auf eine Positionsanzeige in unabhängigen Fenstern oder Frames verzichten.

Anforderung 20 (Fehlertoleranz)

Potentielle Fehlbedienungen des Benutzers sollten berücksichtigt werden.

Benutzer machen typischerweise von Zeit zu Zeit auch Fehler bei der Bedienung eines Systems (zum Beispiel die Eingabe von Werten außerhalb eines definierten Wertebereiches). Auf diese sollte das System so vorbereitet sein, dass folgende Reaktionen vermieden werden:

- Das System stürzt ab oder „hängt“ endlos.
- Das Ergebnis liefert eine unverständliche (Fehler-)Meldung.
- Die Werte werden ohne Hinweis durch das System angepasst.
- Das Ergebnis wird ohne Hinweis fehlerhaft berechnet.

Stattdessen sollten hilfreiche Rückmeldungen gegeben werden. Ist die Bedeutung der Benutzereingabe auch fehlerhaft noch völlig eindeutig, so kann in

²Insbesondere bei JAVA-Applets wird diese Anforderung sehr schnell verletzt. Diese sind also mit Vorsicht einzusetzen und ggf. mit einer entsprechenden Warnung zu versehen.

Einzelfällen auch ein automatischer Korrekturvorschlag erfolgen, den der Benutzer zu bestätigen hat. Dies sollte aber nur dann geschehen, wenn die genaue Eingabe für den Lerninhalt irrelevant ist.

Beispiel: Bei der Benutzung einer Suchfunktion können ggf. die Metazeichen „%“ und „*“ automatisch an die Bedürfnisse der benutzten Implementierung der Suchfunktion angepaßt und gleichbehandelt werden, da es für den Lerninhalt nicht wichtig ist, ob der Benutzer nun die eine oder die andere Form benutzt.

Über allem steht natürlich das Prinzip der Fehlervermeidung, d.h. die Gestaltung der Einheiten und die verwendeten Texte sollten Fehler so weit wie möglich von vorneherein vermeiden helfen.

Anforderung 21 (Problem toleranz)

| |
|---|
| Potentielle (technische) Probleme und Einschränkungen berücksichtigen |
|---|

Die Praxis zeigt, dass bei bestimmten Formaten und Medienformen (auch trotz teilweiser Standardisierung) nicht immer von einem reibungslosen Einsatz auszugehen ist. So ist zum Beispiel bei der Verwendung von Ton zu bedenken, dass manche Benutzer aus technischen oder aber aus organisatorischen Gründen (Störung der Nachbararbeitsplätze) diesen unter Umständen nicht oder nur eingeschränkt nutzen können. Prinzipiell in diese Kategorie fallen unserer Erfahrung nach auch:

- Proprietäre Textformate (Postscript, PDF, RTF, DOC)
- Programminhalte auf Client-Seite (Java, Java-Script, ActiveX)
- Audio (wav, MIDI, mpeg1-L3, auch als „mp3“ bezeichnet)
- Video (MPEG, AVI, MOV)
- andere aktive Inhalte (Macromedia Flash, Schockwave)
- Neue Formatierungen seit HTML 3.0 (Layer, CSS2, XSL, XML)

Solche Formate können natürlich eingesetzt werden, aber sie sollten dann redundant ausgelegt werden oder entbehrlich sein. D.h. das Modul muss auch unter Verzicht auf solche Formate benutzbar sein. Im Einzelfall können auch Mindestanforderungen (z.B. „Java-fähiger Browser“) definiert werden, wenn auf bestimmte Inhalte nicht verzichtet werden kann.

Redundanz bedeutet hier, dass zum Beispiel in Audio-Files enthaltene Texte zusätzlich auch als (HTML-)Text abrufbar sein müssen.

Die Problemtoleranz sollte möglichst sogar so weit gehen, dass Benutzer die Wiedergabe von Grafiken und Bildern ganz abschalten können, aber anstelle deren dann hinreichend klare Kurzbeschreibungen der Bildinhalte in Textform vorfinden.

6 Hintergrund: Ergonomie im Kontext von Bildschirmarbeit

6.1 Die Bildschirmarbeitsverordnung

Am 20. Dezember 1996 wurde die Bildschirmarbeitsverordnung (BildscharbV) in Kraft gesetzt. Sie ist die Umsetzung der EG-Richtlinie 90/270/EWG „über die Mindestvorschriften bezüglich der Sicherheit und des Gesundheitsschutzes bei der Arbeit an Bildschirmgeräten“ in das nationale Recht der Bundesrepublik Deutschland. Der Geltungsbereich dieser Verordnung bezieht sich laut §2 auf „Beschäftigte, die gewöhnlich bei einem nicht unwesentlichen Teil ihrer normalen Arbeit ein Bildschirmgerät benutzen“ und schließt nicht nur derartige Arbeitsplätze im Bürobereich mit ein, sondern alle Arbeitsplätze mit Bildschirm, die über Einrichtungen zur Erfassung von Daten verfügen und mit Software, Zusatzeinrichtungen und zum Betreiben erforderlichen Elementen ausgestattet sind.

Die Arbeitgeber sind damit verpflichtet, „bei Bildschirmarbeitsplätzen die Sicherheits- und Gesundheitsbedingungen insbesondere hinsichtlich einer möglichen Gefährdung des Sehvermögens sowie körperlicher Probleme und psychischer Belastungen zu ermitteln und zu beurteilen“ (§3). Die zu überprüfenden Sachverhalte umfassen insbesondere Bildschirmgerät und Tastatur, sonstige Arbeitsmittel, die Arbeitsumgebung und den täglichen Arbeitsablauf, sowie das Zusammenwirken Mensch-Arbeitsmittel, das vor allem durch die eingesetzte Software bestimmt wird.

Die Entwicklung der auf Bildschirmarbeitsplätze bezogenen internationalen Normen, die europäische Bildschirmrichtlinie und ihre Umsetzung durch die BildscharbV mit dem ArbSchG zielen erkennbar auf einen um präventive Gesichtspunkte wesentlich erweiterten Gesundheitsschutz und eine Analyse der das gesamte Arbeitssystem umfassenden Beanspruchungs- und Gefährdungspotentiale. Vor allem wird Software zunehmend als zentrale Gestaltungskomponente der Arbeitsbedingungen immer größerer Gruppen der arbeitenden Bevölkerung erkannt und hat ihre ergonomische Qualität (Gebrauchstauglichkeit) durch rahmenhaft, aber nicht im Detail festgelegte Prüf- und Meßverfahren nachzuweisen. In den Erläuterungen der BildscharbV wird darauf hingewiesen, daß die Verordnung „einen flexiblen Rahmen dar(stellt), der Spielraum für an die Situation der Betriebe angepaßte Arbeitsschutzmaßnahmen läßt“. Damit besteht gerade wegen des Verzichts auf eine detaillierte und verbindliche Verfahrensvorschrift vielerorts Bedarf an einer kompakten übersicht der zu berücksichtigenden Erkenntnisse sowie der grundlegenden

Verfahrensschritte zur Überprüfung insbesondere der in den Dialogrichtlinien ISO EN 9241-10 formulierten Prinzipien. Die vorliegende Darstellung kann hierzu vorerst nicht mehr als eine erste Orientierungshilfe bieten.

Die speziell auf Software bezogenen Anforderungen der BildschirmarbV ergeben sich aus deren Anhang in einem „Zusammenwirken Mensch-Arbeitsmittel“ überschriebenen Abschnitt der Bildschirmarbeitsverordnung (S.11):

- 20. Die Grundsätze der Ergonomie sind insbesondere auf die Verarbeitung von Informationen durch den Menschen anzuwenden.
- 21. Bei der Entwicklung, Auswahl, Erwerb und Änderung von Software sowie bei der Gestaltung der Tätigkeit an Bildschirmgeräten hat der Arbeitgeber den folgenden Grundsätzen insbesondere im Hinblick auf die Benutzerfreundlichkeit Rechnung zu tragen:
 - 21.1 Die Software muß an die auszuführende Aufgabe angepaßt sein.
 - 21.2 Die Systeme müssen den Benutzern Angaben über die jeweiligen Dialogabläufe unmittelbar oder auf Verlangen machen.
 - 21.3 Die Systeme müssen den Benutzern die Beeinflussung der jeweiligen Dialogabläufe ermöglichen sowie eventuelle Fehler bei der Handhabung beschreiben und deren Beseitigung mit begrenztem Arbeitsaufwand erlauben.
 - 21.4 Die Software muß entsprechend den Kenntnissen und Erfahrungen der Benutzer im Hinblick auf die auszuführende Aufgabe angepaßt werden können.
- 22. Ohne Wissen der Benutzer darf keine Vorrichtung zur qualitativen oder quantitativen Kontrolle verwendet werden.

Die Anwendung ergonomischer Prinzipien auf die menschliche Informationsverarbeitung findet sich mit vielen Differenzierungen in den internationalen Normen der ISO-Reihe 9241. Mit den Forderungen der Punkte 21.1 – 21.4 werden in verkürzter Form die in ISO 9241 Teil 10 formulierten sieben „Grundsätze der Dialoggestaltung“ angesprochen:

- **Aufgabenangemessenheit** (Forderung 21.1)
- **Selbstbeschreibungsfähigkeit** (Forderung 21.2)
- **Steuerbarkeit** (Forderung 21.3)
- **Fehlertoleranz** (Forderung 21.3)
- **Erwartungskonformität**
- **Individualisierbarkeit** (Forderung 21.4)
- **Lernförderlichkeit**

Diese wichtigen Grundsätze wurden bereits in Kap. 4 (Seite 43)) im Detail und anhand von Beispielen näher erläutert.

6.2 Systemische Betrachtungen zur Bildschirmarbeit

Bereits in der EG-Rahmenrichtlinie (89/391/EWG) und in der Bildschirmrichtlinie ist das „Denken in Arbeitssystemen“ erkennbar, das dem möglichen Vorgehen bei der Analyse der von Bildschirmarbeit ausgehenden Gefährdungen zugrunde liegen sollte. Den multifaktoriellen Beziehungen zwischen Ursachen (Belastungen), Wirkungen (Beanspruchungen) und angemessenen Maßnahmen läßt sich am besten im Rahmen des gedanklichen Modells eines Arbeitssystems nachgehen. In der europäischen Vornorm DIN V ENV 26385 wird dieses wie folgt definiert:

„Das Arbeitssystem beinhaltet das Zusammenwirken von Mensch und Arbeitsmitteln im Arbeitsablauf, um die Arbeitsaufgabe am Arbeitsplatz in der Arbeitsumgebung unter den durch die Arbeitsaufgabe gesetzten Bedingungen zu erfüllen“.

Überdies ist bei Bildschirmarbeit zu berücksichtigen, daß dabei auftretende Beanspruchungen zumeist als vorübergehende oder aber manifeste Befindlichkeitsstörungen auftreten. Dies gilt für Beanspruchungen des Stütz- und Bewegungsapparates, der Augen und solche psychomentaler Art, die unter Präventionsgesichtspunkten in den Ablauf eines möglichen Erkrankungsprozesses einzuordnen sind. Meist handelt es sich weniger um Gesundheitsschäden, sondern um arbeitsbedingte gesundheitliche Gefährdungen und vielfach um unspezifische Beanspruchungen. So können etwa Kopfschmerzen durch andauernde visuelle oder muskuläre Überbeanspruchung hervorgerufen werden, aber ebenso durch psychische Über- oder Unterforderung.

Eine vertiefende Betrachtung des menschlichen Leistungsvermögens und des damit einhergehenden Belastungs-Beanspruchungs-Geschehens mit seinen Rück- und Wechselwirkungen läßt die folgende systemische Darstellung nützlich erscheinen, in der die grundlegenden, bei entsprechenden Untersuchungen zu berücksichtigenden Zusammenhänge zwischen Mensch, Arbeitsaufgabe und - organisational und situativ bedingten – Ausführungsbedingungen dargestellt werden (Abb. 6.1). Jede differenzierte Beanspruchungsanalyse ist somit vor allem als soziotechnische Systemanalyse zu konzipieren und kann sich nicht allein auf die Betrachtung des Arbeitsmittels bzw. die Mensch-Maschine-Schnittstelle beschränken (letztere ist in Abb. 6.1 unter den Bedingungen der Aufgabenausführung subsumiert und nicht explizit dargestellt). Zugleich sind sowohl objektive Einflußfaktoren als auch das subjektive Erleben und die Verarbeitung der Beanspruchungssituation durch die Person zu berücksichtigen. Bezüglich dieser Auffassung besteht in der einschlägigen Literatur weitestgehend Übereinstimmung.

Ebenso macht Abb. 6.1 die Notwendigkeit einer dynamischen Betrachtung der Beanspruchung als eines Prozesses mit positiven und negativen, kürzer und längerfristig wirksamen Beanspruchungsfolgen deutlich, die über entsprechende Rückkoppelungsbeziehungen selbst wieder auf die Eingangsgrößen zurückwirken. Lenprozesse bilden dabei einen wichtigen Aspekt *positiver* Beanspruchungsfolgen und können fortlaufend dazu beitragen, die Beanspruchung im optimalen Bereich zu halten.

Auch in dieser Hinsicht - und bezüglich der damit verbundenen methodischen Schwierigkeiten der Beanspruchungsmessung - besteht in der Literatur prinzipiell Übereinstimmung.

Ansonsten aber werden unterschiedliche Modelle angeboten, an denen man sich auch in Untersuchungen zum Einfluß der Software auf das Beanspruchungsgeschehen zu orientieren hätte. Sie verdeutlichen die Vielfalt möglicher Verfahrensansätze und die Komplexität der dabei zu berücksichtigenden Zusammenhänge (vgl. z.B. [Hacker und Richter 80], [Schönpflug 87], [Wieland-Eckelmann 92], [Burmester et al. 97]).

Zugleich wird erkennbar, daß an ein einfaches „Patentrezept“ zur Beantwortung dieser Thematik nicht zu denken ist. Ebenso wenig läßt sich das Problem auf die Frage reduzieren, ob eher 'objektive' Meßverfahren oder 'subjektive' Bewertungen den angemessenen Zugang bilden. Die adäquateste Antwort dürfte in einem Sowohl-als-auch liegen.

Auf jeden Fall ist [Eilers, Nachreiner und Böning 90] zuzustimmen:

„Die Erfahrungen und Ergebnisse der letzten 30 bis 40 Jahre sollten uns eigentlich gelehrt haben, daß es *den* Indikator psychischer Beanspruchung nicht gibt und mit größter Wahrscheinlichkeit auch nicht geben kann und daß bis heute keine validen *universellen, d.h. tätigkeitsunspezifischen*, Verfahren entwickelt werden konnten, mit deren Hilfe es möglich ist, für jede beliebige Arbeitstätigkeit valide Aussagen über eine zu geringe, angemessene oder erhöhte psychische (Fehl-)Beanspruchung zu erhalten (...). Größere Erfolgsaussichten und größere wissenschaftliche Redlichkeit versprechen dagegen Verfahren, die spezifisch, konzeptadäquat und für bestimmte, begrenzte Gültigkeits- und Anwendungsbereiche konstruiert werden (...).“ (aus [Eilers, Nachreiner und Böning 90], S. 28)

Eine weitere systemische Sicht ergibt sich nicht zuletzt auch bei einer genaueren Analyse der sog. Mensch-Maschine-Schnittstelle (in der Bildschirmarbeitsverordnung unter dem „Zusammenwirken Mensch-Arbeitsmittel“ angesprochen), womit unmittelbar die besonderen Anforderungen an Software-Entwickler veranschaulicht werden.

Trotz ihrer großen und weiter zunehmenden Verbreitung in der Arbeitswelt (vgl. [Bullinger 93]) wird Software selten explizit als Arbeitsmittel entworfen, entwickelt und gehandelt. Das häufige Fehlen systematischer Analysen des zukünftigen Verwendungszwecks und die in der Folge zwangsläufig dürftige Überprüfung der zweckbezogenen Realisierung lassen einen bedenklichen Mangel an ingenieurwissenschaftlichem Vorgehen ahnen. Die Angaben zu Leistungsspektrum, Leistungsverhalten und Gebrauchsgütern in Softwarebeschreibungen erinnern im Vergleich zu denen anderer technischer Produkte eher an Prototypen und Machbarkeitsstudien. Der selbe Mangel läßt sich bezüglich der Benutzungsschnittstelle und der Überprüfung ihrer software-ergonomischen Qualitäten konstatieren.

An ein Arbeitsmittel sind prinzipiell andere Anforderungen zu stellen als an Technik, die für das Forschungslabor oder den privaten Gebrauch entwickelt wird:

6.2 Systemische Betrachtungen zur Bildschirmarbeit

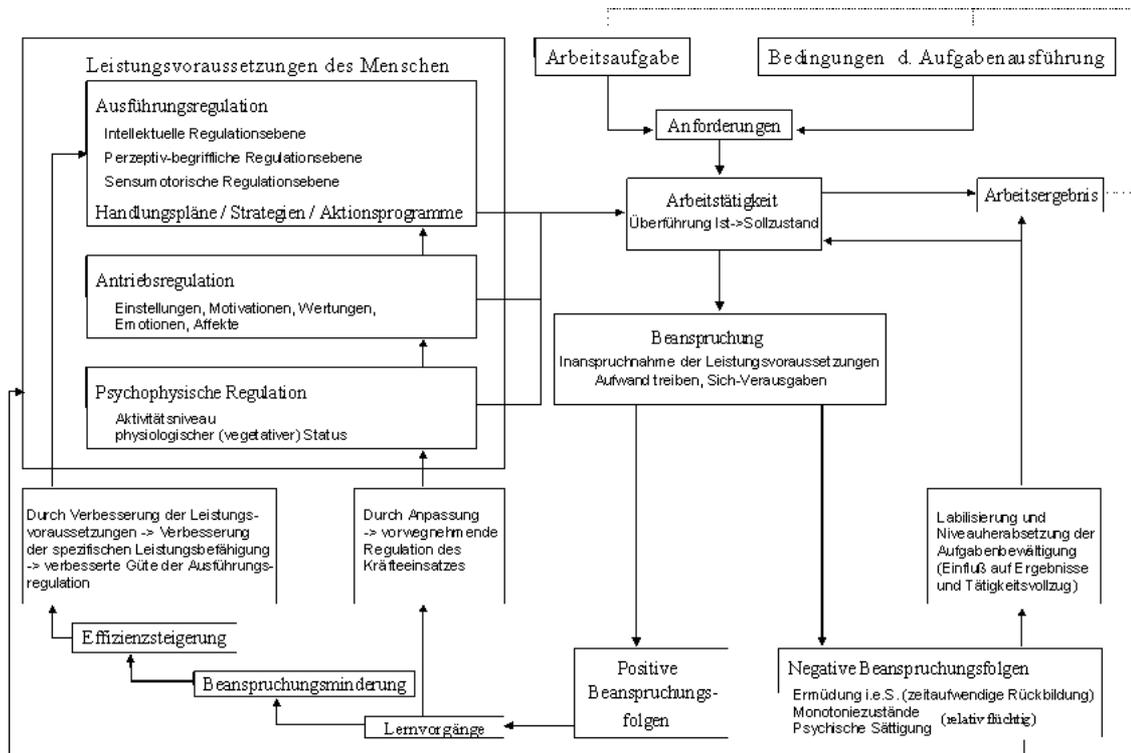


Abbildung 6.1: Systemkomponenten des Beanspruchungsprozesses (nach Plath 1976)

- Arbeitsmittel müssen grundsätzlich vom Großteil der arbeitenden Bevölkerung benutzbar und nicht auf die Bedürfnisse und Besonderheiten vorselegierter Teilpopulationen (z.B. ihrer eigenen Entwickler) ausgerichtet sein. Aber genau so scheint es sich zu verhalten, wenn über Schwierigkeiten älterer Leute mit bestimmten Interaktionsformen (vgl. [Egan 88]) oder die Entwicklungen neuer Softwaresysteme berichtet wird, bei denen der eigene Anwendungswunsch der Entwickler offensichtlich das treibende Motiv ist, wie [Oberquelle 91] (S. 54) an den Akteuren im Bereich von „computer supported cooperative work“ (kurz CSCW) kritisiert.
- Arbeitsmittel müssen gleichmäßige, fehlerarme Dauerleistungen ermöglichen. Es reicht weder aus, die bloße Ausführbarkeit einer Tätigkeit mit einer bestimmten Software nachzuweisen, um sie als Arbeitsmittel zu qualifizieren, noch die schnellere Ausführbarkeit mit einer Software im Vergleich zur langsameren mit einer anderen, um die erstere als besseres Arbeitsmittel auszuweisen. [Nachreiner und Mesenholl 93] kritisieren unter Verweis darauf, daß Software-Ergonomen Ausführungszeit und fehlerlose Aufgabenbewältigung als Bewährungskriterium in Kurzzeitversuchen bevorzugen, die Fixierung der software-ergonomischen Forschung auf Machbarkeitsstudien und Rekordversuche.

Zwar liegt eine der Besonderheiten des Arbeitsmittels Software in den eng begrenzten sensorischen und motorischen Anforderungen, die die Ein-/Ausgabeschnittstelle dem Benutzer zu bieten hat, aber das rechtfertigt noch lange nicht die einseitige Ausrichtung der Software-Ergonomie auf die von [Bonitz 88] kritisierte Übertragung vorwiegend kognitionspsychologischer Laborergebnisse. Die Einseitigkeit der Anforderungen an der Ein-/Ausgabeschnittstelle sollte vielmehr ein Grund mehr sein, sowohl durch eine sorgfältige klassisch-ergonomische Arbeitsmittelgestaltung, als auch durch die Schaffung von nur teilweise rechnergestützten Misch Tätigkeiten (vgl. Bildschirmarbeitsverordnung §5) eine größere Anforderungsvielfalt zu gewährleisten, bzw. dem Benutzer die Entwicklung individueller Entlastungsstrategien zu ermöglichen.

Eine der auffälligsten Besonderheiten des Arbeitsmittels Software beschrieb Dzida bereits 1985 folgendermaßen:

„... Wir benutzen Sprachen, um Rechner zu steuern und erhalten sprachliche Rückmeldung; also kann man verlangen, daß der Rechner dem Benutzer Auskünfte gibt, die ihn bei der Planung unterstützen. Hierin besteht einer der entscheidenden technischen Fortschritte in der Werkzeuggeschichte des Menschen. Es gilt, in arbeitswissenschaftlich verträglicher Weise hieraus Nutzen zu ziehen“ ([Dzida 85] Seite 435).

Über die ‚Auskunfts-fähigkeit‘ des Werkzeuges Software hinaus besteht eine weitere seiner Besonderheiten in seiner ‚Erinnerungs-fähigkeit‘. Einfache Beispiele für deren Nutzung in modernen Standardsoftwaresystemen stellen

die Wiederholfunktionen dar, die heute in fast jedem komfortableren Zeichen- oder Desktop-Publishing-System enthalten sind. Sie benutzen softwaretechnisch gesehen dem selben Mechanismus wie die sog. ‚Undo-Funktion‘, die das Anullieren des letzten Dialogschritts (nach Art und Umfang vom System definiert) erlaubt.

Ist in der Undo- oder der Wiederholfunktion der Umfang des Dialogschritts vom System her definiert und von Benutzerseite normalerweise nicht zu beeinflussen, so verfügen vor allem komplexe CAD- und Grafiksysteme über ausgefeiltere Funktionen ihres Werkzeuggedächtnisses. Bei sog. ‚Makrotracern‘ speichert das System auf Wunsch das Vorgehen während einer oder mehrerer Arbeitseinheiten, damit der Benutzer sich diese Aufzeichnung seiner Dialogsequenzen nachträglich ansehen, sie bearbeiten (z.B. beliebig unterteilen, kombinieren, optimieren etc.) und als vorgefertigte individuelle Prozedur (Makro) zu einem späteren Zeitpunkt wieder einsetzen kann.

Eine solche neue Qualität des Werkzeuggedächtnisses gewann mit der enormen Verbreitung des Interaktionsmodus der sog. ‚direkten Manipulation‘ (kurz DM) – der heute die Mehrzahl der Bildschirmarbeitsplätze charakterisiert - zunehmend an Bedeutung. Diese Form der Interaktion zerlegt den Umgang mit dem Dialogsystem in viele kleine, anschauliche Manipulationsschritte (Objekt auswählen, markieren, Funktion auswählen, auslösen, ausgelöste Veränderung bestätigen oder rückgängig machen), und entbindet den Benutzer so von der Notwendigkeit, eine unanschauliche Kommandosprache erlernen zu müssen. Aber die Aufsplitterung in viele einfache Arbeitsschritte, die dem Anfänger, der die Prinzipien der DM begriffen hat, einen raschen Einstieg in jedes Softwaresystem mit DM-Modus ermöglicht, langweilt und behindert unter Umständen den geübten Benutzer besonders bei der Erledigung komplexer Aufgaben. Durch die Möglichkeit, aus der Spur (trace) seiner vielen kleinen Manipulationen mittels Aufzeichnung, Optimierung und Benennung kleine eigene Programme (Makros) auf Vorrat zu gewinnen, kann der Benutzer seine Produktivität erhöhen und ermüdende Routinetätigkeiten vermeiden.

Die Entwicklung von Software-Systemen ist ohne Schnittstellendefinitionen und Modellvorstellungen über Struktur und Ablauf der Kommunikation nicht möglich. Entsprechend werden in der software-ergonomischen Literatur seit längerem Modelle unterschiedlicher Detailliertheit diskutiert, die einer besseren Übersicht über Zusammenhänge zwischen wichtigen Komponenten des Mensch-Rechner-Systems dienen (vgl. z.B. [Rouse 80], [Dzida 83], [Oberquelle et al. 83], [Rohr und Tauber 84], [Streitz 85], [Koch und Reiterer 91], [Wieland-Eckelmann 92]).

Auf den ersten Blick gibt es an Bildschirm-Arbeitsplätzen nur *eine* Mensch-Maschine-Schnittstelle: Für den Benutzer ist die Ein-/Ausgabeschnittstelle die eigentliche ‚Bedienungsoberfläche‘, an der er z.Z. zumeist über eine mehr oder weniger schreibmaschinenähnliche Tastatur - sowie u.U. andere Hilfsmittel wie z.B. Maus, Lichtgriffel, Digitalisiertablett - seine Anweisungen an das System eingibt, und an der ihm die Ausgaben des Systems - Arbeitsergebnisse, weitere Eingabeaufforderungen (Prompts), Fehlermeldungen u.a. - in der Regel auf einem Bildschirm - sowie bei Bedarf über weitere Ausgabegeräte wie Drucker und Plotter, Kommunikationsverbindungen zu anderen Stellen

in und außerhalb des Betriebs, etc. - zur Verfügung stehen. Die ergonomischen Qualitäten dieser ‚Benutzungsschnittstelle‘ werden sowohl von der Hard- als auch der Software - und deren Zusammenspiel - bestimmt. Im vorliegenden Zusammenhang beziehen wir uns aber wesentlich auf die Einflüsse seitens der Software.

Aus ergonomischer Sicht ist diese Mensch-Maschine-Schnittstelle im Hinblick auf die Tätigkeit und die Aufgaben des Benutzers näher zu beschreiben. Schnittstellen-Modelle haben eine *pragmatische* Funktion: Sie sollen eine möglichst präzise Beschreibung und Abgrenzung der verschiedenen Gestaltungs- und Bewertungsfragen erlauben, die sich bezüglich der Eignung der eingesetzten Software ergeben. Unter diesem Aspekt ist eine erweiterte Fassung des auch bei der Erarbeitung der DIN- bzw. ISO-Normen verwendeten Modells einer Arbeitsgruppe der International Federation of Information Processing (IFIP WG 6.5) nach wie vor ein nützliches Hilfsmittel.

Das ursprünglich von H. Williamson vorgeschlagene und in deutschsprachiger Fassung von Dzida 1983 [Dzida 83] vorgestellte Modell ist ein inzwischen auf breiter Basis akzeptierter Versuch, die verschiedenen Gestaltungsaspekte von Benutzungsschnittstellen möglichst unabhängig voneinander zu beschreiben (auch wenn sie sich nicht völlig unabhängig voneinander gestalten lassen und für den Benutzer selbst einen einheitlichen Zusammenhang der ihm im Dialog zur Verfügung gestellten Handlungsmöglichkeiten, Arbeitsschritte und Ergebnisse bilden).

Das ursprüngliche sog. IFIP-Modell unterscheidet vier Aspekte der Mensch-Rechner-Interaktion (und daraus abgeleitete ‚Schnittstellen‘):

- Ein-/Ausgabe (E/A)
- Dialogabwicklung
- Gebrauch von Werkzeugen (oder Informationen)
- Einbettung des Mensch-Rechner-Systems in die Organisationsumgebung

6.2.1 Ein-/Ausgabe-Schnittstelle

Nach den Erläuterungen von Dzida ([Dzida 83] S.6) definieren „Regeln für die Eingaben des Benutzers und die Ausgaben des Software-Systems... diesen Teil der Benutzerschnittstelle.

- *Eingaberegeln*: Es wird festgelegt, auf welche Weise Zeichen eingegeben werden oder womit die Schreibmarke auf dem Bildschirm positioniert wird (...). Ferner wird festgelegt, wie Aufträge beschrieben sind und womit sie dem System erteilt werden können (zum Beispiel mittels Kommandonamen oder Funktionstasten).
- *Ausgaberegeln*: Es wird festgelegt, auf welche Weise Daten und Werkzeuge dem Benutzer dargestellt werden. Die vom System ausgegebenen Daten sollen gruppiert und formatiert dargestellt werden... Geeignete

Kodierungsformen sind auszuwählen: Durch Ortskodierung kann man Feldern auf dem Bildschirm bestimmte Bedeutungen zuordnen; durch Farbkodierung kann man die Aufmerksamkeit des Benutzers lenken.”

6.2.2 Dialog-Schnittstelle

Die mit der raschen technischen Entwicklung (schnellere Prozessoren, höhere Speicherkapazitäten, grafikfähige Displays etc.) gewachsenen Möglichkeiten, den Rechner in einer ‚dialog‘-ähnlichen Weise als Arbeitsmittel in unterschiedlichsten Anwendungsbereichen einzusetzen, sind einer der Hauptgründe für die zunehmende Verbreitung der Informationstechniken und die daraus entstehenden ergonomischen Probleme. Bei der früher üblichen sog. Stapel-Verarbeitung (batch processing) war der Benutzer gezwungen, die an den Rechner delegierten Teile seiner Arbeitsaufgabe vorher schrittweise vollständig durchzuplanen und in eine programmähnliche Liste von Befehlen zu übersetzen (was eine Reihe EDV-bezogener Spezialkenntnisse voraussetzte); mit beträchtlicher zeitlicher Verzögerung erhielt er die erwarteten Ergebnisse oder auch nur eine Liste von Fehlermeldungen, die ihn dazu zwangen, den ganzen Prozeß der Planung noch einmal oder mehrfach zu durchlaufen.

Es gehört zu den gesicherten arbeitswissenschaftlichen Erkenntnissen, daß gerade die im ‚Dialogbetrieb‘ mögliche rasche Rückmeldung von Teilergebnissen und/oder Fehlern Lernprozesse erleichtert, und daß eine sehr weit vorausgreifende und auf nur vorgestellte Zwischenresultate angewiesene Planung die kognitive Kapazität vieler Menschen überfordert. Insofern erscheint es gut begründet, daß die Gestaltung der Dialog-Schnittstelle im Zentrum softwareergonomischer Interessen steht und in den ISO-Normen 9241 eine wichtige Rolle spielt.

Nach den Erläuterungen von Dzida kennzeichnen

„Regeln, die den Dialog zwischen Benutzer und Software-System festlegen,... die Dialogschnittstelle. Es wird beispielsweise geregelt, wie der Benutzer sich Hilfen durch das System geben lassen kann, um ein Eingabefeld verstehen zu lernen oder den Anwendungszweck eines Werkzeugs erklärt zu bekommen. Es wird auch geregelt, wie der Benutzer Aufträge erteilen kann, beispielsweise erst nach Aufforderung durch das System oder nach Auswahl eines Menüs. Ferner wird geregelt, wie der Benutzer die Auftragsbearbeitung durch das System beeinflussen, zum Beispiel unterbrechen, fortführen oder abbrechen kann. Man kann Dialogformen benennen, die dem Benutzer das Lernen oder Planen erleichtern (...). Auch die Regeln zur Behandlung von Fehlersituationen sind ein Problem der Gestaltung der Dialogschnittstelle.“ [Dzida 83]

6.2.3 Werkzeug-Schnittstelle

Die Werkzeugschnittstelle bezieht sich auf die Arbeitsaufgabe des Benutzers im eigentlichen Sinne, indem sie Arbeitsgegenstände in ihrem jeweiligen Bearbeitungsstadium und die zur Bearbeitung verfügbaren Instrumente („Werkzeuge“, „Software-“, „Maschinen“) symbolisch repräsentiert.

Wingert [Wingert 83] hat auf die Problematik der Werkzeug-Metapher im vorliegenden Zusammenhang hingewiesen. Das beim Werkzeuggebrauch i.e.S. unmittelbar sinnlich erfahrbare (und hierüber die Handhabung regulierende) Erlebnis der kontinuierlichen Gegenstandsveränderung ist eher im Rechner ‚versteckt‘ und kann allenfalls - je nach Umfang der rückgemeldeten (Zwischen-) Ergebnisse - gedanklich rekonstruiert werden. Software-ergonomische Ansätze bilden z.T. den Versuch, die dadurch steigenden Anforderungen an das Abstraktionsvermögen zu kompensieren.

Diese Überlegungen Wingerts bilden u.E. aber auch schon eine Begründung dafür, das vorgeschlagene Modell dadurch zu erweitern, daß Werkzeuge und Arbeitsgegenstand getrennt thematisiert werden - wir kommen hierauf so gleich noch zurück.

Nach den Erläuterungen von Dzida (siehe [Dzida 83] S.6) charakterisieren „Regeln, die den Zugriff des Benutzers auf Software-Werkzeuge und Daten bestimmen,...die Werkzeugschnittstelle. Den Benutzer interessiert, welche Dienste die Werkzeuge anbieten, wie gut die Werkzeuge aufeinander abgestimmt sind, ob beispielsweise für das Formulieren und Versenden einer Nachricht derselbe Editor zur Verfügung steht wie für das Formulieren eines Monatsberichts. Für den Benutzer stellt der Computer eine Menge von kombinierten Werkzeugen dar; die Kombinationsprinzipien bestimmen die Werkzeugschnittstelle (zum Beispiel das Zusammenfassen von Werkzeugen zu einer Prozedur). Ferner kann für den Benutzer wichtig sein, voreingestellte Parameter zu ändern oder überhaupt Änderungen an Werkzeugen vorzunehmen.“

Mit der analytischen Unterscheidung verschiedener Aspekte der Benutzungsschnittstelle wird - wie Dzida in [Dzida 83] bei der Vorstellung des Modells schon hervorhob - die Möglichkeit ins Auge gefaßt, schrittweise zu Vereinbarungen (Standards) über jeden der angesprochenen Bereiche zu kommen, anhand derer sich aus software-ergonomischer Sicht verbindliche (Mindest-) Anforderungen unabhängig von bestimmten Anwendungen festlegen lassen. Die Entwicklung und Implementation anwendungs-unabhängiger (oder -neutraler) Benutzungsschnittstellen könnte dazu beitragen, daß Benutzer sich nur in begrenztem Maße auf eine bestimmte Software spezialisieren müssen (und dadurch einer inhaltliche Verarmung rechnergestützter Büro- und Verwaltungstätigkeiten entgegengewirkt wird). Denn nur bei einer möglichst einheitlichen Benutzungsschnittstelle für beispielsweise so unterschiedliche Anwendungen wie Textverarbeitung, Datenbankanwendungen, Tabellenkalkulation und Geschäftsgraphik hält sich der erforderliche Lern- und Übungsaufwand so weit in Grenzen, daß der professionelle Umgang ein und desselben Benutzers (etwa eines qualifizierten Sachbearbeiters) mit *allen* diesen Programmen möglich erscheint. Überdies ergeben sich selbst beim hauptsächlichlichen Umgang mit nur einer Anwendung beträchtliche Transfereffekte, falls doch einmal gelegentlich auch andere Anwendungen genutzt werden sollten.

Mit anderen Worten: die von Ulich (siehe [Ulich 91]) als besonders wesentlich herausgestellten Prinzipien einer ‚differentiellen‘ und ‚dynamischen‘ Arbeitsgestaltung können software-seitig durch die Entwicklung anwendungs-unabhängiger Schnittstellen unterstützt werden. Es ist dann im weiteren eine

Frage der Organisation, insbesondere bezüglich der horizontalen Aufgaben- und der vertikalen Vollmachtenteilung, inwieweit diese Optionen auch mit Einführung der modernen Informationstechnologie genutzt werden.

6.2.4 Organisationsschnittstelle

Die Einbeziehung der Organisationsschnittstelle in das vorliegende Modell macht deutlich, daß die Bedeutung des organisatorischen Kontexts der Mensch-Rechner-Interaktion nicht übersehen wird.

Dzida erläutert: „Regeln, die den Zusammenhang der Arbeitsaufgaben des Benutzers mit den Arbeitsaufgaben anderer Benutzer bestimmen, kennzeichnen die Organisationsschnittstelle: zum Beispiel Arbeitsteilung, Dienstweg, Regeln der Kooperation. Die Werkzeuge eines Software-Systems sollten auch auf die übrigen Werkzeuge des Benutzerarbeitsplatzes abgestimmt sein; die Regeln dieses Zusammenhangs kennzeichnen ebenfalls die Organisationsschnittstelle, wie die Kopplung eines ‚electronic mail systems‘ mit der hausinternen Post.“ ([Dzida 93] S. 6).

6.2.5 Ein Vorschlag zur Ergänzung des IFIP-Modells

Wir hatten schon angedeutet, daß uns die an der Werkzeug-Schnittstelle des IFIP-Modells zu konstatierende Vermengung des Arbeitsgegenstandes (Daten) und der Mittel zu seiner Bearbeitung (Werkzeuge) unbefriedigend erscheint. Es ist verständlich, daß es für Informatiker zunächst keiner Unterscheidung zwischen Werkzeugen und Daten bedurfte, solange beide aus mehr oder minder formatierten Zeichenketten bestanden. Mit der Verbreitung der objektorientierten Programmierung wird die notwendige Unterscheidung auch für die Informatik erkennbar, und für den Benutzer bei direktmanipulativen graphischen Oberflächen deutlich sichtbar.

Aus ergonomischer und arbeitspsychologischer Sicht erscheint es uns nicht nur nützlich, sondern geradezu zwingend,

- die software-bedingte Darstellung des *Arbeitsgegenstandes* bei Gestaltung und Evaluation auch gesondert zu berücksichtigen: An der Ein-/Ausgabe-Schnittstelle ist die Darstellung des Arbeitsgegenstandes und des an ihm sichtbaren Bearbeitungsfortschritts seit jeher - gegenüber der Darstellung der verfügbaren Dialoghilfsmittel und der Werkzeuge - von zentraler Bedeutung (und es ist anzunehmen, daß z.B. sein vorübergehendes Verschwinden vom Bildschirm in bestimmten Dialogsituationen erhebliche Befürchtungen auslösen kann, die mit der Angst vor Datenverlust zu tun haben und mit der Immaterialität seiner Existenz zusammenhängen). Ein entsprechend erweitertes IFIP-Modell zeigt Abb. 6.2.

Anhand der Relation Arbeitsgegenstand - Werkzeug ist es darüber hinaus ebenso wichtig,

- auf die Begrenzungen des Gestaltungskonzepts ‚anwendungs-unabhängiger‘ Schnittstellen hinzuweisen. Die gesamte Geschichte der Werkzeugentwicklung zeigt deutlich, daß stets zunehmend spezifischere, genau auf die Tätigkeit und den Arbeitsgegenstand abgestimmte Werkzeuge entwickelt werden, die sich meist schon auf den ersten Blick von anderen Werkzeugen unterscheiden, und deren professioneller Gebrauch oft nur über längere Zeit erlernt werden kann. Gleiches gilt für die Gestaltung von Software: Wenn sie ergonomisch wirklich auf die Belange ganz bestimmter Tätigkeiten abgestimmt sein soll, wird sie sich sicherlich nicht auf die Gestaltung anwendungs-unabhängiger Schnittstellen beschränken können. Allenfalls kann es darum gehen, dort wo es sinnvoll und möglich erscheint, anwendungsunabhängige Gesichtspunkte zu berücksichtigen, dann aber - hierauf aufbauend - die Gestaltungserfordernisse als anwendungs- und tätigkeitsspezifisch zu begreifen und in ergonomisch adäquaten Lösungen zu realisieren.

Das erweiterte Modell soll außerdem verdeutlichen, daß die Arbeitsaufgabe gewissermaßen die ‚Schnittstelle‘ zwischen Organisation und Benutzer bildet (vgl. auch [Volpert 87] S. 14), da im wesentlichen über sie seine formalen Beziehungen zur Organisation (und seine Stellung als Arbeitnehmer) bestimmt sind. In vergleichbarer Weise bildet das im Rechner - bzw. im Rahmen der Mensch-Maschine-Interaktion - erzeugte Arbeitsergebnis schließlich wieder eine Schnittstelle zur Organisation: Es kann z.B. von Bedeutung für Entlohnung und Aufstiegschancen sein (und dies erklärt vermutlich auch eine Reihe von Befürchtungen und möglichen Stressoren bei der Einführung der neuen Technologie).

Ebenso entscheidet die Organisation über die weitere Verwertung des Ergebnisses, z.B. auch darüber, inwieweit es zugleich zur ‚Schnittstelle‘ zwischen dem Benutzer und anderen Mitarbeitern des Unternehmens wird, und auf diese Weise Bestandteil der Kommunikations- und Kooperationsbeziehungen des Unternehmens (bzw. soziotechnischen Systems) ist, bzw. in diese Beziehungen wesentlich eingreift. Dies heißt aber auch, daß durch die Art und Weise, in der Software den Austausch der Arbeitsergebnisse unterstützt oder behindert, ein bedeutsames Qualitätsmerkmal der Organisationsschnittstelle definiert wird.

Diesem Gesichtspunkt wird erst in den letzten Jahren vermehrt Aufmerksamkeit zuteil, nachdem inzwischen ‚computer-supported cooperative work‘ (CSCW) oder auch ‚Groupware‘ als besondere Anwendungsprogramme auf den Markt drängen. Entsprechend zeichnet sich ein neuer Forschungsschwerpunkt der Software-Ergonomie ab (vgl. z.B. schon [Oberquelle 91], [Friedrich 93]). Für die Zukunft prognostizierte deshalb bereits Naffah ([Naffah 91] S. 23) die wachsende Bedeutung einer ‚Gruppen-Maschine-Schnittstelle‘:

„There is the need to develop new kinds of user interfaces, based on the idea of a ‚group-machine interface‘ rather than the older concept of an interface between a single user and a machine; the interface now has to provide for communication between a group of users and the electronic system.”

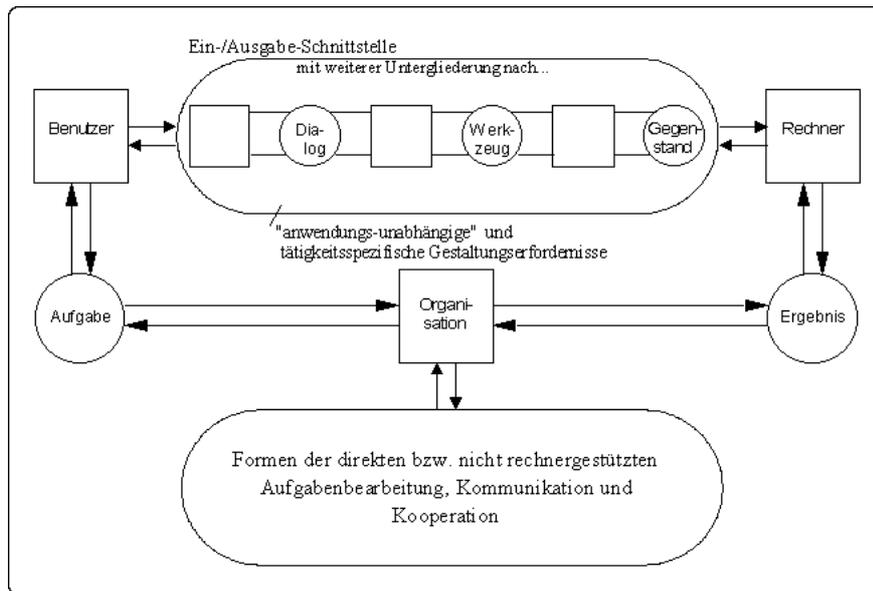


Abbildung 6.2: Das erweiterte IFIP-Schnittstellenmodell

In noch höherem Maße gilt dies natürlich für Systeme, die verschiedene Darstellungsmedien und -hilfsmittel integrieren und deren gemeinsame, synchrone oder asynchrone Nutzung auch über große Distanzen hinweg ermöglichen sollen.

Aus den hier dargestellten Gründen werden in dem von uns vorgeschlagenen erweiterten IFIP-Modell auch alle Formen der direkten bzw. nicht rechnergestützten Aufgabenbearbeitung, Kommunikation und Kooperation dem rechnergestützten Ablauf als Parallele gegenübergestellt. Die besonderen Mischungsverhältnisse beider Arbeitsformen, und ihre Verträglichkeit (Kompatibilität), bilden den Kern der Frage nach der menschengerechten Gestaltung von solchen soziotechnischen Systemen, zu deren Komponenten ‚dialogfähige‘ Rechner gehören.

6.3 Software-ergonomische Gestaltungsgrundsätze und Normen

In der software-ergonomischen Literatur finden sich eine ganze Reihe unterschiedlicher, wenn auch in Teilen übereinstimmender Gestaltungsgrundsätze und Richtlinien, deren Differenzierungsgrad und theoretische Begründungen erheblich voneinander abweichen. Dadurch ist es schwierig, die verschiedenen Ansätze aufeinander abzubilden (vgl. z.B. [Balzert 87], [Ulich et al. 91], [VDI 88]).

Daneben haben die meisten großen Hersteller inzwischen eigene Richtlinien (sog. Styleguides) entwickelt, die vor allem das in der Praxis allmählich erworbene ergonomische Know-how zusammenfassen und die Wahrschein-

lichkeit erhöhen, daß auch bei der Programmierung unterschiedlicher Anwendungen - und der Zuarbeit externer Software-Entwickler - nach einheitlichen Prinzipien gestaltete und bedienbare Benutzungsoberflächen entstehen (vgl. als Übersichtsreferat [Eberleh 94]). Diese herstellereigenen Richtlinien betreffen allerdings in erster Linie graphische Benutzeroberflächen zur Dialogtechnik der sogenannten ‚direkten Manipulation‘ bei Standardsoftware für den Bürobereich, wie sie dem im angloamerikanischen Raum als ‚knowledge worker‘ bezeichneten Benutzer angeboten wird. Überdies dienen sie vorwiegend kommerziellen Interessen, indem sie einerseits eine unverkennbare ‚corporate identity‘ an der Benutzungsoberfläche sichtbar machen, und andererseits den Benutzer verschiedener Anwendungen auf die Produktfamilie ein und desselben Herstellers festzulegen versuchen.

Soweit derartige Styleguides bereits in ihnen entsprechende Programmierwerkzeuge (sog. Toolkits) transformiert sind, ist das zugrundeliegende Know-how sogar schon in solchem Maße vergegen- und verselbständigt, daß Programmierer es schematisch - und im Extremfall ohne eigene ergonomische Kenntnisse oder Überlegungen - in neue Anwendungsprogramme einbringen könnten.

Wir gehen im folgenden schwerpunktmäßig nur auf jene Gestaltungsgrundsätze ein, die bereits Eingang in die internationale Normierungsarbeit gefunden haben und deshalb dem bisher erreichten Minimalkonsens bezüglich ‚gesicherter arbeitswissenschaftlicher Erkenntnisse‘ entsprechen, deren Berücksichtigung Softwareentwickler nachzuweisen haben. Auf weitere Grundsätze und Gestaltungsrichtlinien kommen wir danach nur exemplarisch und zusammenfassend zu sprechen.

6.4 Die ISO-Normen 9241 als software-ergonomisches Referenzkonzept

In der Internationalen Organisation für Standardisierung (ISO) werden seit Mitte der achtziger Jahre software-ergonomische Normen entwickelt und diskutiert, die als Referenzkonzept für software-ergonomische Gestaltungsansätze herangezogen werden können, zumal sie sich in ihrer Entwicklung auch aufeinander beziehen und wechselseitig beeinflussen. Eine Übersicht zum aktuellen Stand der Bildschirm-Arbeitsplätze betreffenden, sowohl auf Hard- als auch auf Software bezogenen ISO-Normenreihe 9241 „Ergonomische Anforderungen für Bürotätigkeiten mit Bildschirmgeräten“ gibt Tab. 6.1.

Auf die Software-Ergonomie i.e.S. beziehen sich die Teile 10-17 der ISO-Normenreihe 9241. Dabei definiert Teil 11 das Konzept der *Gebrauchstauglichkeit* (usability); man vergleiche auch DIN 66 050 zur Definition von ‚Gebrauchstauglichkeit‘. Die Teile 12 (Präsentations-Prinzipien) und 10 (Dialog-Prinzipien) enthalten übergeordnete Gestaltungsgrundsätze, die auch in den spezielleren Normen der Teile 13-17 Berücksichtigung finden. Nicht zuletzt ist unter ergonomischen Gesichtspunkten auch Teil 2 zu beachten, da er Aussagen zur Gestaltung der Arbeit und der Aufgaben des Benutzers macht, und damit einer zu eingegrenzten Sichtweise vorbeugt.

| Teil | Titel | Status | Hierarchie |
|------|--|---------|------------|
| 1 | General introduction (in turnusmäßiger Überarbeitung) Allgemeine Einführung | IS | A0 |
| 2 | Guidance on task requirements (in Überarbeitung) Anforderungen an die Arbeitsaufgaben - Leitsätze | IS | A0 |
| 3 | Visual display requirements Anforderungen an visuelle Anzeigen | IS | B; C |
| 4 | Keyboard requirements Anforderungen an Tastaturen | DIS | B; C |
| 5 | Workstation layout and postural requirements Anforderungen an Arbeitsplatzgestaltung und Körperhaltung | DIS | A1; B |
| 6 | Environmental requirements Anforderungen an die Arbeitsumgebung | DIS | A1; B |
| 7 | Display requirements with reflections Anforderungen an visuelle Anzeigen bezüglich Reflexionen | CD | C |
| 8 | Requirements for displayed colors Anforderungen an Farbdarstellungen | DIS | B; C |
| 9 | Requirements for non-keyboard input devices Anforderungen an Eingabegeräte - ausgenommen Tastaturen | DIS | A1; B |
| 10 | Dialogue principles (-> DIN 66234 T.8) Grundsätze der Dialoggestaltung | IS | A1 |
| 11 | Guidance on Usability Angaben zur Benutzbarkeit | DIS | A0 |
| 12 | Presentation of information (-> DIN 66234 T.3+5) Informationsdarstellung | DIS | A1 |
| 13 | User guidance Benutzerführung | DIS | A1 |
| 14 | Menu dialogues Dialogführung mittels Menüs | DIS | B |
| 15 | Command dialogues Dialogführung mittels Kommandosprachen | DIS | B |
| 16 | Direct manipulation dialogues Dialogführung mittels direkter Manipulation | DIS (f) | B |
| 17 | Form-filling dialogues Dialogführung mittels Bildschirmformularen | DIS | B |

Tabelle 6.1: ISO Normenreihe ‚Ergonomic requirements for office work with visual display terminals (VDTs)‘ - ISO 9241 Part 1-18 (unter CE 29241 als europäische Norm übernommen).
Abkürzungen zum Status bedeuten (bei zunehmender Vorläufigkeit des bis jetzt vorliegenden Textes):
IS=International Standard, DIS=Draft International Standard, CD=Committee Draft, WD=Working Draft

[Çakir und Dzida 97] haben dafür plädiert, die Normen der Reihe ISO 9241 in Anlehnung an das Modell der Normen zur Maschinensicherheit nach Hierarchie-Ebenen zu klassifizieren, um auf diese Weise Umfang und Gewicht der Anforderungen einzelner Normen besser zu verdeutlichen. Die letzte Spalte von Tab. 6.1 zeigt eine derartige Zuordnung. Dabei enthält die oberste Ebene A (A0; A1) gewissermaßen Grundnormen, die wichtige Zielvorgaben beinhalten. Ebene B betrifft Normen, die das Erreichen entsprechender Ziele spezifizieren, während auf Ebene C mehr oder weniger reine Produktnormen angesiedelt sind. Schäfer & Çakir (1997) haben in diesem Zusammenhang auch die weitere Überarbeitungsbedürftigkeit der gesamten Normenreihe verdeutlicht (ein Punkt, auf den hier nicht weiter einzugehen ist).

Als weitere, für die Software-Ergonomie wichtige ISO-Normen sind zu erwähnen:

- ISO / IEC 9126:Software product evaluation: Quality characteristics and guidelines for their use;
- ISO 12 119:Information processing - Software packages - Quality requirements and testing;
- ISO / IEC 10 741-1:Information technology - User system interfaces for text and office systems - Dialogue interaction / Part 1: Cursor control for text editing;
- ISO / IEC 11 580:Information technology - User system interfaces for text and office systems - Names and descriptions of objects commonly used in the office environments.

Schließlich ist auf allgemeine – nicht speziell auf Bildschirmarbeitsplätze oder Software bezogene – Normen hinzuweisen, die wegen ihrer grundlegenden Bedeutung in jedem Falle beachtet werden sollten; insbesondere sind dies:

- ISO 6385: Ergonomic principles in the design of work systems (sozusagen die Grundnorm der ergonomischen Gestaltung von Arbeitssystemen);
- ISO 10075: Ergonomic principles related to mental work load (Parts 1-3).

Im Unterschied zu *technischen* Normen (die sich auf geforderte Produkteigenschaften beziehen) ist für *ergonomische* Normen ihr *Richtliniencharakter* (guidelines) bezeichnend. Bei technischen Normen, die sich z.B. auf physikalische meßbare Eigenschaften beziehen, können exakte Grenzwerte angegeben werden, für die Meßmethoden bekannt und eventuell Meßapparaturen bereits vorhanden sind. Die Begründung für die Notwendigkeit einer bestimmten Eigenschaft, die Relevanz des Grenzwerts, die Gültigkeit der Meßmethode und das Konstruktionsprinzip der Meßapparatur sind in diesem Fall alle aus der gemeinsamen und anerkannten ingenieurwissenschaftlichen Basis abgeleitet.

Im Falle ergonomischer Normen werden die ‚weicheren‘ Richtlinien bevorzugt, weil unter den Experten - von den wissenschaftlichen Grundlagen, über die Ableitung zu fordernder Eigenschaften, bis hin zu deren Differenzierung und der Entwicklung von Erfassungsmethoden - zu jedem einzelnen Schritt noch mehrere unterschiedliche, z.T. auch gegensätzliche Auffassungen vorherrschen. Solche Meinungsvielfalt unter Wissenschaftlern und Technikern ist ein typisches Merkmal einer jungen Disziplin und tritt auch in traditionellen Wissenschaften immer wieder dort auf, wo das gesicherte Grundwissen erweitert wird.

Normen in Gegenstandsbereichen, die sich noch im Zustand der Forschung und Entwicklung befinden, beruhen daher vielfach auf dem Konsens der in den Normungsausschüssen versammelten Experten. Da sich die Verankerung der Empfehlungen nicht automatisch aus entsprechenden wissenschaftlichen Theorien ergibt, werden in ergonomischen Normen häufig Begründungen für die getroffenen Entscheidungen mitgeliefert und die Anwendbarkeit der Regelung wird dabei oft durch Beispiele erläutert. Beides führt dazu, daß ergonomische Normen nicht selten als zu umfangreich und damit schlecht handhabbar, unscharf und interpretierbar angesehen werden. Die selben Argumente werden auch als Kritik gegenüber hersteller- oder anwendereigenen Style-Guides vorgebracht.

Vielen Technikern, die an ‚harte‘, beispielsweise physikalisch begründete Normen gewöhnt sind, erscheint eine Lösung des Problems durch radikale Vereinfachung der benutzten Begriffe und die gleichzeitige Reduzierung auf die dem sogenannten ‚gesunden Menschenverstand‘ zugänglichen Inhalte möglich. Die Resultate eines solchen Vorgehens zeigen sich in den z.T. eklektischen und trivialen Bemerkungen zu Themen wie ‚usability‘ und ‚Benutzerfreundlichkeit‘ in nationalen und internationalen Normen zur Qualitätssicherung.

Abhilfe ist nur dadurch zu schaffen, daß die wissenschaftliche Begründung, die exemplarische Erläuterung und die Darstellung von Erhebungsmethoden von der Norm getrennt beschrieben werden, und die Experten sich - parallel zur Weiterentwicklung der Norm - auf ein abgestimmtes Evaluationsverfahren einigen (das dann in einem eigenen Normenteil dargestellt werden sollte). Ein solches Verfahren würde im Zuge seiner Anwendung zusätzlich dazu beitragen, die noch vorhandenen Lücken im (software-) ergonomischen Grundwissen zu schließen. Mittlerweile entwickelte und diskutierte Evaluationskonzepte machen deutlich, daß auch für solche Normen eine Konformitätsprüfung prinzipiell möglich und sinnvoll ist.

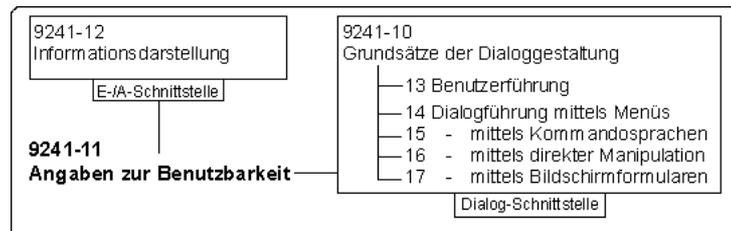


Abbildung 6.3: Die auf Software bezogenen ergonomischen Normen der ISO 9241

Ergonomische Normen enthalten Forderungen unterschiedlicher Stärke: ‚Harte‘ Forderungen (requirements) sind als Soll-Aussagen (shall) formuliert - etwas ist so zu regeln, hat so zu sein. ‚Weiche‘ vertretene Forderungen (recommendations) - abhängig vom aktuellen Diskussions- und Erkenntnisstand - sind demgegenüber als Sollte-Aussagen (should) formuliert. Darüber hinaus berücksichtigen *bedingte* Forderungen (conditional requirements), daß es im Kontext menschlicher Arbeit sinnvoll sein kann, unterschiedliche Rahmenbedingungen (in einer Wenn-Klausel formuliert) zu beachten (z.B. „Wenn Interaktionen reversibel sind und die Aufgabe es zuläßt, soll der jeweils letzte Dialogschritt rückgängig gemacht werden können“).

Da Normung nicht nur den aktuellen Stand arbeitswissenschaftlicher Erkenntnisse berücksichtigt, sondern auch auf Konsens beruht, werden die Mehrzahl software-ergonomischer Normen wohl zunächst vielfach nur Empfehlungscharakter (recommendations) haben, wenngleich abzuwarten bleibt, wie sich in diesem Zusammenhang die EG-Richtlinie 90/270/EWG auswirken wird (vgl. [Çakir und Dzida 97]).

Im folgenden werden die Inhalte der ISO 9241, soweit sie bereits vorliegen und sich auf den hier zur Diskussion stehenden *software*-ergonomischen Kontext beziehen, in einem kursorischen Überblick dargestellt; hinsichtlich einzelner Definitionen und Prüferfordernisse ist aber auf die Normentexte selbst zu verweisen. Die strukturellen Beziehungen zwischen den diesbezüglichen Teilen der ISO 9241 veranschaulicht Abb. 6.3.

Dementsprechend beschreibt Teil 11 mit den geforderten Angaben zur Gebrauchstauglichkeit die übergeordneten Gesichtspunkte und Zielvorgaben, denen die anderen Teile funktional ‚zuarbeiten‘. Teil 12 regelt die Informationsdarstellung an der Ein-/Ausgabe-Schnittstelle. Die Teile 10 und 13 – 17 beschreiben Anforderungen an die Dialog-Schnittstelle, wobei Teil 10 übergeordnete Gesichtspunkte enthält, die unabhängig von jeweils spezifischen Dialogtechniken Geltung beanspruchen. In gewissem Umfang gilt dies allerdings auch für Teil 13, da eine geeignete Benutzerführung ebenfalls bei jeder Art der Dialogtechnik gewährleistet sein muß.

Die im folgenden kurz näher erläuterten Normen folgen insgesamt einem einheitlichen Aufbau (Gliederung). Einem einleitenden Teil, der die jeweilige Norm ins Gesamtkonzept der Normenreihe und deren Adressatenkreis einordnet, folgen die Abschnitte

- 1. Geltungs-, Anwendungsbereich (Scope);
- 2. Bezüge zu anderen Normen (Normative References);
- 3. Definitionen der für die jeweilige Norm relevanten Begriffe und Gestaltungsprinzipien (Definitions);
- 4. Grundsätzliches zur Anwendung der Norm (Application);
- 5ff. Empfehlungen (Recommendations) im Detail, gegliedert unter Bezug auf die in Abschnitt 3. gegebenen Definitionen;
- Anhänge (bis zu drei), die ausdrücklich als ‚informativ‘ deklariert sind. Sie beziehen sich auf: A. Verwendete Literatur; B. Exemplarische Lösungen; C. Checklisten.

6.4.1 Anforderungen an die Gebrauchstauglichkeit - Leitsätze (ISO 9241-11)

In Teil 11 wird „usability“ als die Qualität der Interaktion zwischen Benutzern und anderen Teilen des Arbeitssystems definiert. Das entspricht am ehesten dem deutschen Begriff der *Gebrauchstauglichkeit*.

Das *Arbeitssystem* setzt sich nach Teil 11 aus der durchzuführenden Aufgabe, den Merkmalen des Benutzers, den Merkmalen der technischen Ausrüstung und denen der sozialen und physischen Umgebung zusammen, wobei jede einzelne Komponente Einfluß auf die Gebrauchstauglichkeit eines Softwaresystems nehmen kann. Dementsprechend kann die Gebrauchstauglichkeit auch nur unter systematischer Berücksichtigung jeder dieser Komponenten und ihres Zusammenspiels geprüft werden.

Weiter wird festgehalten, daß die Überprüfung der Gebrauchstauglichkeit eines Softwareprodukts - als dem bestimmenden Faktor für die Arbeit an Bildschirmgeräten - sinnvoll nur *im Rahmen seines Nutzungskontextes* stattfinden kann, welcher aus dem Benutzer, seiner Aufgabe, seinen Arbeitsmitteln und der Arbeitsumgebung gebildet wird. Ferner sind das zu prüfende Softwareprodukt und die Ziele seines Einsatzes zu beschreiben. In den Erläuterungen zur weiteren Untergliederung der einzelnen Komponenten wird dabei ausdrücklich vor der Gefahr gewarnt, in eine technikzentrierte Begrifflichkeit zu verfallen, also Aufgaben des Benutzers und Ziele des Softwareeinsatzes nur als Funktionen des Produkts wiederzugeben, anstatt sie als Schritte zur Zielerreichung darzustellen.

Als Kriterien für die Gebrauchstauglichkeit eines Softwareprodukts und die mit ihm zu erreichende Arbeitsqualität werden in ISO 9241 Part 11 Effektivität, Effizienz und Zufriedenheit vorgeschlagen:

Gebrauchstauglichkeit Das Ausmaß, in dem ein Produkt durch bestimmte Benutzer in einem bestimmten Nutzungskontext genutzt werden kann, um bestimmte Ziele effektiv, effizient und mit Zufriedenheit zu erreichen.

Arbeitsqualität in einem genutzten Arbeitssystem: Das Ausmaß, in dem bestimmte Ziele in einem bestimmten Arbeitssystem effektiv, effizient und mit Zufriedenheit erreicht werden können.

Effektivität Die Genauigkeit und Vollständigkeit, mit der Benutzer ein bestimmtes Ziel erreichen.

Effizienz Der im Verhältnis zur Genauigkeit und Vollständigkeit eingesetzte Aufwand, mit dem Benutzer ein bestimmtes Ziel erreichen.

Zufriedenheit Beeinträchtigungsfreiheit und Akzeptanz der Nutzung

(zitiert aus EN 29241-11, 3.1-3.5).

Legt man diese Konzeption zugrunde, so hängt es von der jeweils konkreten Konstellation der Nutzungsbedingungen ab, ob eine Software die an sie gestellten Anforderungen erfüllt, denn die allgemeinen Bewährungskriterien müssen in Abhängigkeit von den aktuellen Nutzungsbedingungen spezifiziert werden. Ein solches Vorgehen erscheint u.U. zwar aufwendig, aber es ist methodisch begründet, durchaus durchführbar und aussichtsreich.

Einige der in den Beispielen von ISO 9241-11 genannten Maße (wie Anzahl der Versuche bis zur fehlerfreien Ausführung oder Zeitmessungen zur Aufwandsbestimmung) erscheinen zwar eher für kurzzeitige Laboruntersuchungen an einfachen Aufgaben geeignet, und weniger als Korrelat der Beanspruchung des Benutzers, die über einen längeren Zeitraum des regelmäßigen Arbeitens mit einem bestimmten Softwareprodukt entsteht. Da aber auch auf andere Maße (z.B. Fehlbeanspruchungen und ihre Folgen) und Erfassungsmöglichkeiten ausdrücklich hingewiesen wird, steht es dem Untersucher frei, auch weitergehende Vorstellung auf der Grundlage von ISO 9241-11 zu entwickeln.

Obwohl im Unterschied zu technischen (auch z.B. hardwarebezogenen) Normen ein bestimmtes Prüfverfahren nicht vorgeschrieben wird (und sinnvollerweise auch nicht werden sollte), sind in ISO 9241 - 11 hinreichend klare Anforderungen formuliert:

„Eine Beschreibung der Maße der Gebrauchstauglichkeit enthält Zielwerte oder tatsächliche Werte der Effektivität, Effizienz und Zufriedenheit, und zwar für die jeweiligen Nutzungskontexte. Normalerweise braucht man mindestens ein Maß jeweils für Effektivität, Effizienz und Zufriedenheit. Weil die relative Bedeutung dieser Komponenten der Gebrauchstauglichkeit sowohl vom Nutzungskontext abhängt als auch vom Zweck, für den die Gebrauchstauglichkeit zu beschreiben ist, gibt es keine allgemeine Regel dafür wie Maße ausgewählt oder kombiniert werden sollen.

Auswahl und Detaillierungsgrad jedes Maßes hängen von den Zielen der an der Messung beteiligten Parteien ab. Die Bedeutung, die jedes Maß im Verhältnis zu den Zielen hat, sollte beachtet werden. Beispielsweise haben bei seltener Nutzung Maße für Lernen oder Wiedererlernen große Bedeutung.

Wenn es nicht möglich ist, objektive Messungen für Effektivität und Effizienz zu erhalten, dann können subjektive Messungen, die auf dem Urteil des Benutzers beruhen, als Indikatoren für Effektivität und Effizienz dienen“ (zitiert aus EN 29241-11, 5.3.1).

In drei Anhängen (A, B und C) bietet ISO 9241-11 dem Praktiker zwar keine Rezepte, aber eine Beispielsammlung für die Spezifikation des Nutzungskontexts, für Maße der Gebrauchstauglichkeit und die Spezifikation von Gebrauchstauglichkeitsanforderungen.

6.4.2 Ein-/Ausgabeschnittstelle: Informationsdarstellung (ISO 9241-12)

Teil 12, den wir der Ein-/Ausgabeschnittstelle zuordnen, gehört zu den bereits am weitesten entwickelten software-ergonomischen Teilen der Normenreihe. In ihm werden sowohl generell anwendbare als auch konditionale Richtlinien und Empfehlungen für die Anordnung und Darstellung von Informationen auf dem Bildschirm gegeben. Er beschäftigt sich mit optisch erfassbaren Informationen auf zeichen- und grafikorientierten Bildschirmen. Im Gegensatz zum Teil 14 (Dialogführung mittels Menüs) sind akustische Ein-/Ausgaben ausdrücklich von der Betrachtung ausgeschlossen.

Die Gliederungsgesichtspunkte, auf die sich die Empfehlungen beziehen, umfassen drei Hauptbereiche:

- 1. Organisation der Information;
- 2. Graphische Objekte;
- 3. Techniken zur Codierung der Information (nicht zu verwechseln mit der Erstellung von Programmcode, worauf die Norm ausdrücklich hinweist).

Für jeden dieser Hauptbereiche werden die Empfehlungen weiter differenziert:

- **1. Organisation der Information**
 - Fenster
 - Bereiche
 - Eingabe- / Ausgabe-Bereich
 - Gruppen
 - Listen
 - Tabellen
 - Bezeichner (Labels)
 - Felder
- **2. Graphische Objekte**
 - Allgemeine Empfehlungen
 - Positionsmarken und Zeigeinstrumente (Cursor)
- **Techniken zur Codierung der Information**

- Allgemeine Empfehlungen
- Alphanumerische Codierungen
- Abkürzungen (für alphanumerische Codes)
- graphische Codierung
- Farbcodierung
- Codierung mittels anderer Visualisierungstechniken

Die Übersicht macht erkennbar, daß für diesen Teil der Normenreihe Querbezüge zu so gut wie allen anderen software-ergonomischen Teilen zu beachten sind (vgl. auch das im vorangehenden dargestellte Schnittstellenmodell).

Die Hinweise, die in Teil 12 zur Farbgestaltung gegeben werden, sind ausdrücklich auf ihren Einsatz zur Hervorhebung und Kategorisierung von Informationen beschränkt. Für die weiterreichenden Grundsätze der Farbgestaltung wird ausdrücklich auf Richtlinien zum Einsatz von Farbe (ISO 9241-8) verwiesen, während so wichtige Probleme wie z.B. die minimale Zeichengröße beim Arbeiten mit einer Zoom- oder Lupenfunktion nicht hinreichend angesprochen werden. Obwohl die Informationsdarstellung für zeichen- *und* grafikorientierte Bildschirme geregelt werden soll, sucht man Hinweise für die Kombination beider Darstellungsweisen noch vergeblich (z.B. eine zeichenorientierte Software, die in einem Fenster einer grafikorientierten Oberfläche läuft). In Zukunft wären Regeln für sinnvolle oder zu vermeidende Kombinationen wünschenswert.

Auf Unterschiede in der Informationsdarstellung, die lediglich zur Absicherung eines bestimmten firmenspezifischen Erscheinungsbilds dienen sollen (wie sie in sog. Styleguides festgelegt sind), erstreckt sich der Geltungsbereich von Teil 12 nicht.

6.4.3 Dialogschnittstelle: Grundsätze der Dialoggestaltung (ISO 9241-10)

Für diesen Teil der ISO 9241, der bereits als Norm Gültigkeit erlangt hat, ist zu betonen, wieviele Übereinstimmungen mit DIN 66234 Teil 8 bestehen. Tatsächlich haben die deutschen Teilnehmer der WG 5 (Working Group) den Teil 8 ihrer nationalen Norm als am weitesten entwickeltes und auch international viel beachtetes Konzept frühzeitig in die Diskussion eingebracht.

In der deutschen Übersetzung heißt ISO 9241-10 "Grundsätze der Dialoggestaltung", wie der bekannte Teil 8 der DIN, und umfaßt neben den von dort weitgehend übernommenen Grundsätzen

- Aufgabenangemessenheit,
- Selbstbeschreibungsfähigkeit,
- Steuerbarkeit,
- Erwartungskonformität

- Fehlerrobustheit

zusätzlich die beiden Grundsätze

- Individualisierbarkeit und
- Erlernbarkeit.

Die beiden letzteren Gestaltungsgrundsätze waren früher schon in der deutschen Normendiskussion im Gespräch (siehe [Triebel, Wittstock und Schiele 87]). Über diese zusätzlichen Anforderungen an gute Dialoggestaltung ließ sich damals kein Einverständnis (vor allem zwischen Wissenschaftlern und Herstellern) erzielen. Ihre Aufnahme in die internationale Norm dokumentiert u.E. ein durch moderne Möglichkeiten der Dialogtechnik gefördertes Verständnis für den Nutzen und die Notwendigkeit ergonomischer Arbeitsmittelgestaltung in den neuen Technologien. Diese Sichtweise wird auch durch dadurch bestätigt, daß Teil 10 bei der Abstimmung über seine Transformation in eine gültige internationale Norm eine überwältigende Zustimmung fand. Das zeigte sich auch im Auftrag für eine allgemeine Akzeptanzuntersuchung bei Software-Entwicklern (vgl. [Beimel 93]), die ebenfalls eine beträchtliche Zustimmung zur Notwendigkeit und Nützlichkeit von ISO 9241-10 zu Tage förderte.

Andererseits erscheint es uns aufgrund zahlreicher Praxiserfahrungen wichtig darauf hinzuweisen, daß die Aufnahme des Gestaltungsprinzips Individualisierbarkeit doch auch erhebliche Probleme und Gefahren in sich birgt, denen bei der Softwarebeurteilung besondere Aufmerksamkeit geschenkt werden sollte. Generell sind hier u.E. zwei Problembereiche hervorzuheben:

Die angebotenen Möglichkeiten zur Anpassung der Benutzungsoberfläche an individuelle Bedürfnisse und Vorlieben können (1) dazu führen, daß einzelne Benutzer Veränderungen vornehmen, die nicht nur im ergonomischen Sinne Verschlechterungen darstellen, sondern u.U. sogar gesundheitliche Gefährdungen und Beeinträchtigungen mit sich bringen, z.B. wenn Farbkombinationen gewählt werden, die zu einer erhöhten Belastung der Adaptationsfähigkeit der Augen führen. Auf diese Problematik wird auch in der Normenreihe ISO 9241 verschiedentlich bereits hingewiesen (vgl. z.B. Teil 12 am Ende von Abschnitt 1. Geltungsbereich). Wo immer möglich müßten daher Software-Entwickler nur solche Individualisierungsmöglichkeiten zulassen, die sich im Rahmen ergonomischer Standards bewegen.

Zum anderen besteht die Gefahr, daß (2) Software-Entwickler unter Verweis auf eine sehr weitgehende ‚Individualisierbarkeit‘ ihres Produkts ihre Verantwortung für die Anpassung an die spezifischen Bedingungen des Nutzungskontexts an die Benutzer selbst abgeben, d.h. diesen eine Aufgabe übertragen, die nur unter erheblichem Zusatzaufwand zu lösen ist. In der Regel zahlt sich dieser Aufwand für den Benutzer nur aus, wenn er die Möglichkeit hat, die von ihm bevorzugten Einstellungen – in Analogie z.B. zu Druckformaten bei Textprogrammen – auf einfache Weise dauerhaft abzuspeichern (abrufbar zu halten), und dies nach *mehreren* Gesichtspunkten, nämlich

- personenbezogen (was in vernetzten Systemen u.U. schon ausgeschlossen wird);
- gegenstandsbezogen (z.B. Protokoll; Präsentation; Forschungsbericht);

sowie

- situationsbezogen (z.B. Phase der Erstellung eines Dokuments; Endredaktion).

All diese Möglichkeiten werden bisher in den seltensten Fällen angeboten, und das Kriterium einer prinzipiell weitgehenden *Individualisierbarkeit* wird auf diese Weise zu einer ‚Hintertür‘, durch die sich Software-Entwickler ihrer Verantwortung für anwendungs- und benutzerspezifische Produkte entziehen können.

6.4.4 Benutzerführung (ISO 9241-13)

Ähnlich wie Teil 12 bezieht sich auch diese Norm auf Gesichtspunkte, die unabhängig von spezifischen Dialogtechniken Geltung beanspruchen. Im allgemeinen Vorspann von Teil 13 wird zudem ausdrücklich darauf hingewiesen, daß besonders die Richtlinien aus Teil 12 (Informationsdarstellung) auch im Rahmen der verschiedenen Arten, Benutzerunterstützung anzubieten, zu berücksichtigen sind.

Ansonsten enthält Teil 13 neben generellen Empfehlungen auch Aussagen zu

- Eingabeaufforderung (Prompts);
- Rückmeldung (Feedback);
- Statuserkennung (Status);
- Fehlermanagement (Error Management);
- Online-Hilfe (On-line Help);

und wird insgesamt von der Absicht getragen, dem Benutzer ein Maximum an Transparenz und Sicherheit im Umgang mit einem Dialogsystem zu geben.

Als theoretische Grundlage für seine Aussagen verfügt Teil 13 in seinem Anhang über eine Fehlertaxonomie und ein Prozeßmodell zur Fehlerentstehung, -entdeckung, -erklärung und -behebung.

Zusätzliche, nicht direkt auf den Dialog bezogene Hilfestellungen für den Benutzer - wie das Angebot von Online-Tutorien oder der Dokumentation im System - sind vom Geltungsbereich ausgenommen.

6.4.5 Dialogführung mittels Menüs (ISO 9241-14)

Dieser Teil der ISO 9241 gehört zu den umfangreichsten Teilen der Normenreihe, denn neben den Richtlinien und Empfehlungen (die allerdings meist konditional sind) enthält er auch ein ausführliches und in sich konsistentes Glossar der benutzten Fachausdrücke, zahlreiche Hinweise auf Hintergrundliteratur, die in einem speziellen Index den einzelnen Gestaltungshinweisen zugeordnet sind, sowie eine Checkliste, mit der der Schnittstellendesigner das jeweils erreichte Verhältnis von anwendbaren zu umgesetzten Regeln feststellen kann.

Die Hauptgliederungspunkte des Teils 14 (mit zahlreichen weiteren Unteraspekten) beziehen sich auf: (1) Menü-Struktur, (2) Menü-Navigation, (3) angebotene Optionen und (4) Menü-Darstellung. Dabei wird zu Fragen, wie Gruppierung und Reihenfolge der Menüs im Verhältnis zum häufigsten Arbeitsablauf stehen, ebenso Stellung genommen, wie zum Einsatz der verschiedenartigen Ein-/Ausgabemedien in Menüs (z.B. Tastatur, Funktionstasten, Cursortasten, Zeigeinstrumente und Sprachein-/ausgabe). Die Regeln sind dabei so universell gehalten, daß auch andere eventuell erst noch in der Entwicklung befindliche Ein- / Ausgabe-Medien nicht ausgeschlossen werden müssen. Andererseits wird betont, daß viele Regelungen sich auf Sprachen beziehen, bei denen die Lese- bzw. Schreibkonventionen sich auf ‚von links nach rechts‘ beziehen.

Dem Anwender der wird Norm erläutert, wie er feststellen kann, welche Regeln in seinem speziellen Anwendungsfall zu benutzen sind, und er wird ausdrücklich darauf hingewiesen,

- daß die Umsetzung der Gestaltungshinweise die genaue Kenntnis der Aufgabe und Präferenzen des Benutzers sowie die Beherrschung der einzusetzenden Technologie voraussetzt; und
- daß nur eine empirische Prüfung unter Beteiligung repräsentativer Endbenutzer die Erreichung der angestrebten Qualitäten seines Produkts belegen kann.

Die Fülle der Richtlinien und Empfehlungen entspricht dem Umfang des Regelungsbedarfs, weil mit Hilfe der mittlerweile recht einfach zu benutzenden Werkzeuge zur Gestaltung der Benutzungsoberfläche mittels Menüs sich dem Designer eine solche Vielfalt an Kombinationsmöglichkeiten zur Auswahl stellt, daß dabei die Entstehung schlechter Lösungen weitaus wahrscheinlicher ist als die guter, sofern ihm keine entsprechenden Leitlinien zur Verfügung stehen.

6.4.6 Dialogführung mittels Kommandosprachen (ISO 9241-15)

Dieser Teil der Normenreihe enthält Empfehlungen (z.T. in konditionaler Form) zur kommandosprachlichen Dialogführung, auch akustischer Art, und eventuell in Kombination mit anderen Dialogtechniken. Die Hauptgliederungspunkte beziehen sich auf: (1) Struktur und Syntax, (2) Darstellung von Kommandos, (3) Eingabe / Ausgabe sowie (4) Rückmeldung und Hilfe.

Auch in diesem Teil werden ein Glossar mit den wichtigsten Definitionen und Fachausdrücken, eine Literaturliste mit Verweisen auf den wissenschaftlichen Hintergrund der einzelnen Aussagen, sowie Untersuchungsvorschriften zur Feststellung der Anwendbarkeit der Regeln im Einzelfall und zur Prüfung ihrer Umsetzung angegeben.

Die Beschreibung des Geltungsbereichs erfolgt aber eher durch eine negative Abgrenzung gegenüber anderen Formen der Dialogsteuerung durch den Benutzer (z.B. Menüauswahl, Funktionstasten etc.) und hebt besonders auf die Einhaltung der vom jeweiligen System vorgeschriebenen Syntax und Reihenfolge der kommandosprachlichen Dialogführung und der damit verbundenen Gedächtnisbelastung ab. Sie legt somit fest, daß alle unmittelbar verständlichen Piktogramme oder weitgehend natürlichsprachliche Abfragesprachen, die dem Benutzer kaum Gedächtnisleistungen abverlangen, nicht in den Zuständigkeitsbereich dieses Teils der Norm fallen.

Die Frage, ob Mnemonics und Abkürzungen, wie sie häufig anstelle von Menüauswahlen angeboten werden, einen Übergang zu einer Kommandosprache darstellen - ein Eindruck, der sich immer dann aufdrängt, wenn man solche Abkürzungen (short cuts) fortlaufend zur Überbrückung mehrerer Menüstufen eingeben kann - reduziert sich damit auf den Beitrag zur Gedächtnisentlastung. Kurz gesagt, eingängige Mnemonics und short cuts mit einer leicht zu erschließenden Syntax fielen nicht unter Teil 15, sondern nur solche, die das Gedächtnis des Benutzers ähnlich belasten wie eine komplizierte Kommandosprache. Hier hätte sich u.E. eine Unterscheidung nach Komplexität und Reichweite, sowie die Beschreibung des Übergangsbereichs eher angeboten.

Immerhin wird bei den Hinweisen zur Überprüfung der Normenkonformität - wie in den anderen Teilen der Norm - auf den wesentlich erweiterten Teil 11 „Guidance on usability“ verwiesen.

6.4.7 Dialogführung mittels direkter Manipulation (ISO 9241-16)

In Teil 16 wird der Bereich der Dialogführung geregelt, in dem der Benutzer durch Zeigen, Anklicken, Ziehen u.ä. mehr oder minder unmittelbar Funktionen auf meist grafisch dargestellte Objekte, d.h. Softwarekonstrukte und Datenstrukturen, anwendet. Die Notwendigkeit, zwischen den unterschiedlichen Zuständen der verschiedenen Objekte, sowie in besonderen Fällen auch zwischen den Objekten und ihrer Repräsentation zu differenzieren, wird betont. Dabei können die Objekte wahlweise zwei- oder dreidimensional dargestellt werden, während Benutzungsschnittstellen mit stereoskopischen oder „Virtual-reality“-Darstellungsformen explizit vom Geltungsbereich ausgenommen werden.

Die behandelten Aspekte gliedern sich nach: (1) Allgemeine Empfehlungen (Metaphern; Erscheinungsbild von Objekten; Rückmeldungen; Eingabemittel). (2) Manipulation von Objekten (Zeigen und Auswählen; Ziehen; Größenveränderung / Skalierung; Rotation). (3) Manipulation von Textobjekten (Zeigen und Auswählen; Größenveränderung). (4) Manipulation von Fenstern (Zeigen und Auswählen; Ziehen; Größenveränderung / Skalierung).

Bei der Interaktionsform der „Direkten Manipulation“ treten bestimmte Merkmale besonders in den Vordergrund, die bei anderen Interaktionsformen nur eine untergeordnete Rolle spielen oder gar nicht vorkommen können: Die Tatsache, daß die selben Objekte je nach Systemzustand unterschiedlich auf die selbe Benutzeraktion reagieren (Modusabhängigkeit) hat die Software-Entwickler nach Möglichkeiten suchen lassen, dem Benutzer den Modus, in dem sich ein Objekt gerade befindet, möglichst anschaulich zu verdeutlichen. Eine Technik hierfür ist der Einsatz unterschiedlicher Cursor-Formen (z.B. ein Pfeil für allgemeine Zeige- und Auswahloperationen; ein großes I als Texteingabemarke; oder ein Stern, wenn in einem Hypertextsystem Bereiche berührt werden, zu denen noch weitere Informationen hinterlegt sind, etc.). Dieses Thema wird in Teil 16 auch exemplarisch behandelt.

Da direkte Manipulation als Interaktionsform grafische Benutzungsschnittstellen geradezu voraussetzt, deren Gestaltung häufig durch spezielle Werkzeugsysteme (graphical user interface systems, GUIs) und oft firmenspezifische Regelwerke (Styleguides) erleichtert ist, wird deren Anwendung häufig mit Systemen direkter Manipulation gleichgesetzt. Sie umfassen aber meist auch Vorgaben für andere Interaktionsformen (wie z.B. Menüauswahl). Teil 16 konzentriert sich aber auf solche Gestaltungshinweise, die sich auf die direkte Manipulation beziehen. Für die übrigen Bestandteile von GUIs und den zugehörigen Regelwerken müssen Softwareentwickler sich auf die entsprechenden anderen Teile der Norm stützen.

6.4.8 Dialogführung mittels Bildschirmformularen (ISO 9241-17)

Teil 17 „Form-filling dialogues“ enthält Empfehlungen für die Gestaltung von Bildschirmformularen, wie sie im Rahmen der Erledigung typischer Verwaltungstätigkeiten benutzt werden. Die Bildschirmformulare stellen dabei Felder zur Verfügung, in die neue Daten eingegeben oder in denen bereits vorhandene verändert werden, wodurch entweder eine dem Formular unterliegende Datenbank neu angelegt oder eine bereits vorhandene aktualisiert wird. Zwei unterschiedliche Dialogarten werden berücksichtigt, zum einen die Eingabe von beliebigen Zeichenketten oder ihnen zugeordneten Abkürzungen über die Tastatur, zum anderen die Auswahl möglicher Feldeinträge aus Auswahllisten.

Die Darstellung von Bildschirmformularen auf zeichenorientierten und auf grafikfähigen Bildschirmen wird beschrieben. Für letztere werden auch Elemente graphischer Benutzungsschnittstellen berücksichtigt, die sich wie kleine Bildschirmformulare verhalten, z.B. Dialogboxen, wie sie häufig zur Zusammenstellung von Optionen benutzt werden.

Die Gestaltungshinweise beziehen sich auf (1) den Aufbau und die Strukturierung von Bildschirmformularen, (2) die Dateneingabe, (3) geeignete Rückmeldungsmechanismen über den Aufbau und den Zustand einzelner Felder und der ganzen Datenbank, sowie (4) Werkzeuge für die Informationssuche und die Orientierung in Datenbanken (sog. Navigationsmethoden). Sie sind so ausgelegt, daß sie den Prozeß der Softwareentwicklung vom Entwurf über die Realisierung bis hin zur Überprüfung der Gebrauchstauglichkeit be-

gleiten können. Für letztere wird wiederum auf den Teil 11 und die dort beschriebenen Voraussetzungen und Überprüfungsmethoden verwiesen.

6.5 Sonstige Richtlinien, Guidelines und Styleguides

Mitte der achtziger Jahre waren bezüglich der Benutzungsschnittstelle zwei gegenläufige Entwicklungen zu beobachten. Einerseits erreichten ausgehend von software-ergonomischen Untersuchungen (vgl. DZIDA et al. 1977) Anforderungen an eine angemessene Gestaltung den Status einer nationalen Norm (vgl. DIN 66234 Teil 8). Deren Berechtigung und Umsetzbarkeit wurde besonders von Hard- und Softwareherstellern im Großrechnerbereich heftig in Zweifel gezogen, während sie sich heute unter Experten weltweit beträchtlicher Bekanntheit und Zustimmung erfreut, wie [Beimel 93] in einer internationalen Befragung nachweisen konnten.

Andererseits war für die wirtschaftlich rationelle Erstellung grafischer Benutzungsschnittstellen die Entwicklung von Werkzeugen notwendig und machbar geworden, denn im Gegensatz zu den Masken und Formularen für zeichen- und transaktionsorientierte Anwendungssoftware konnten „graphical user interfaces“ (kurz GUI) für flexible integrierte Büroanwendungen nicht mehr für jedes Anwendungssystem gesondert programmiert werden. Objektorientierte Software-Architekturen erlaubten zudem, die Oberflächenobjekte weitgehend getrennt vom Anwendungssystem zu entwickeln, zu verändern und sie von einem Anwendungssystem zum anderen zu übertragen (Portabilität).

Damit waren die Voraussetzungen für den universellen und anwendungsübergreifenden Einsatz einheitlicher grafischer Benutzungsschnittstellen gegeben. Es blieb aber unklar, wie sie gestaltet werden sollten, denn die technischen Möglichkeiten überstiegen den in den (damals noch umstrittenen) Normen bereits geregelten Gestaltungsspielraum bei weitem.

Während die ersten Systeme mit grafischer und direkt-manipulativer Oberfläche wie der Xerox Star oder die Apple Lisa noch als rundum geschlossene Systeme von der Kundenanalyse über ein Benutzerszenario bis zur integrierten Oberflächen- und Anwendungssoftware durchdacht und erstellt worden waren, befanden sich die mächtigen Werkzeuge zur GUI-Gestaltung nun in den Händen von Anwendungsentwicklern ohne software-ergonomische Vorkenntnisse oder in denen von Schnittstellenentwicklern, die keine ausreichenden Informationen über die Anwendungen besaßen. Die Möglichkeit, Anwendungssoftware und GUI getrennt und arbeitsteilig zu erstellen, schaffte - unterstützt durch die Oberflächenwerkzeuge und ihre Objektbibliotheken voller vorfabrizierter Oberflächenbestandteile (*Widgets* genannte „window objects“ wie Menüleisten, Dialogboxen, Piktogramme etc.) - eine Vielfalt, die unbedingt sinnvoll begrenzt werden mußte, wenn nicht völlig uneinheitliche, unübersichtliche und kaum bedienbare Benutzungsschnittstellen entstehen sollten.

„These problems show the need for standards in the field of user interface design. Progress in this direction is absolutely necessary and should

be supported by all IT companies, but not in the sense that each IT company has its own style guide and the user has the choice to select the right one" (aus [Harker und Eason 92] Seite 419).

In dieser Situation formulierten mehrere Hersteller parallel zur (Weiter-) Entwicklung ihrer Baukästen und Werkzeuge für GUIs sogenannte *Styleguides*, während von seiten einiger Großanwender sogenannte *Guidelines* zusammengetragen wurden.

Die Guidelines sind zumeist neutral gegenüber der Hard- und Softwarebasis formuliert und dienen dazu, bei der Erweiterung, Veränderung oder Portierung bereits vorhandener Anwendungssoftware das einheitliche Verhalten und Erscheinungsbild an der Benutzungsschnittstelle zu wahren. Im Falle der Auswahl von Standardsoftware oder der Vergabe eines Entwicklungsauftrags für Individualsoftware bilden die Guidelines einen wichtigen Bestandteil der Lieferbedingungen. Daher wurden die umfangreichsten und detailliertesten Guidelines, wie im öffentlichen Ausschreibungswesen üblich, von Ministerien abgefaßt (vgl. [NASA 87], [DoD 83]). Die wohl umfassendste Zusammenfassung damals existierender oder in der Entwicklung befindlicher Guidelines gaben [Smith und Mosier 84] im sogenannten MITRE-Report.

Da Guidelines zumeist nicht von Herstellern verfaßt wurden, handelt es sich bei ihnen um reine „Paperware“, d.h. gedruckte Regelwerke oder besser Regelsammlungen, die nicht unbedingt nach einer expliziten Systematik geordnet, sondern mit der Zeit durch Hinzufügen von Regeln und Beispielen gewachsen sind. Dementsprechend wird immer wieder von Problemen im Umgang mit Guidelines berichtet, die als wenig operational und schlecht handhabbar empfunden werden (vgl. [Souza und Bevan 90], [Tetzlaff und Schwartz 91]).

Sollen Guidelines nicht nur dazu dienen, an einem fertigen Softwaresystem das Vorhandensein oder Fehlen bestimmter Oberflächenmerkmale zu identifizieren, sondern auch eine Anleitung für den Schnittstellendesigner im Laufe des Entwicklungsprozesses ergeben, wird von den Software-Entwicklern besonders dann, wenn sie keine Software-Ergonomie-Experten sind, der Interpretationsspielraum der angeführten Regeln und Gestaltungsgrundsätze bemängelt. Vor allem die Frage, ob eine Regel in einer bestimmten Situation angewandt werden muß, angewandt werden kann oder eventuell nicht zutrifft, stellt GUI-Entwickler ohne Spezialkenntnisse und langjährige Erfahrung immer wieder vor große Probleme.

„Guidelines at present are best delivered by a human agent“ empfehlen [Kirakowski und Corbett 90] (S. 23), weil ein Software-Ergonomie-Experte oder -Berater am ehesten geeignet erscheint, die vielen meist nur unvollständig spezifizierten Einflußfaktoren intuitiv zu erfassen und in seinen Empfehlungen zu berücksichtigen. Zwar gibt es schon Versuche, die Beraterrolle auf ein Expertensystem zu übertragen, und das mag in begrenzten Gegenstandsbereichen (hier beispielsweise Farbgestaltung) mit einfachen Regeln auch möglich sein. [Maaß 93] warnt aber davor, schematisches Regelwissen anwenden zu wollen, wo die Ausgangssituation nicht hinreichend analysiert

und die Interdependenzen der einzelnen Regeln nicht ausreichend erforscht sind.

Einen anderen Weg zeigen Hüttner, Wandke & Rätz (1995), die von einer rechnergestützten Unterweisung für Software-Entwickler berichten, in der auf dem Bildschirm außer Beispielen (für gute und schlechte Gestaltung) auch Erläuterungen mit den jeweils relevanten Gestaltungsregeln angeboten werden. Dieses Informations- und Beratungssystem für Anwendungsentwickler (intra) liegt inzwischen auch auf Diskette vor.

Die Tendenz bei den Herstellern von Anwendungssoftware und GUI-Werkzeugen geht aber eher in die Richtung, die Einhaltung von (software-ergonomischen) Gestaltungsrichtlinien durch den Einsatz hierfür eigens entwickelter Werkzeuge zu automatisieren. Diese Tendenz ist nicht unproblematisch, weil der Werkzeugeinsatz allein noch keine gut gestalteten Benutzungsschnittstellen garantieren kann. Bisher sind die Werkzeuge auf das beschränkt, was im anglo-amerikanischen Sprachraum in bekannt unbekümmerter Art und Weise als „Look“ (optisches Erscheinungsbild an der Ein-/Ausgabe-Schnittstelle) und „Feel“ (Verhalten an der Dialogschnittstelle) bezeichnet wird.

Den Bezug zur Anwendung, der beispielsweise auf die Gruppierung von Menüeinträgen oder die Reihenfolge von Dialogabläufen entscheidenden Einfluß hat, muß der Entwickler weitgehend ohne Werkzeugunterstützung in Eigeninitiative herstellen. Treten hier Mängel auf, dann können sie nicht von einer in sich stimmigen Benutzungsschnittstelle kompensiert werden. Für die Analyse und Umstrukturierung der Arbeitsaufgabe (wie z.B. in den VDI-Richtlinien gefordert) bieten die uns bekannten Fenstersysteme, Werkzeugkästen und Styleguides ohnehin keine Unterstützung.

Styleguides wurden immer als Regelwerke gemeinsam mit Werkzeugen entwickelt (in Ausnahmefällen auch nur die Werkzeuge, wie bei „New Wave“ von HEWLETT-PACKARD). In der Minimalversion umfaßt das Regelwerk die Definition der wichtigsten Oberflächenobjekte und die Beschreibung ihrer Modifikations- und Kombinationsmöglichkeiten, wie bei „OSF/Motif“ der Open Software Foundation und „OPEN LOOK“ von SUN, deren Schwerpunkt eindeutig in den Werkzeugen und der Widget-Bibliothek liegt. Das andere Extrem bilden für ihre jeweilige Systemumgebung „Common User Access“ (CUA) von IBM und „The Apple desktop interface“. Sie umfassen jeweils das Regelwerk, die Werkzeuge und software-ergonomische Grundprinzipien. Während das Besondere bei CUA die Tatsache ist, daß es als Teil einer Gesamtarchitektur die Einheitlichkeit der Benutzungsschnittstelle über jede nur denkbare Form von Mischkonfiguration (innerhalb der IBM-Hard- und Software) garantieren soll, liefert Apple für Macintosh auch noch eine Anleitung zur ästhetischen Gestaltung, die von einem speziellen Team von Grafikdesignern zusammengestellt wurde.

Hier soll nicht die Notwendigkeit und Nützlichkeit ästhetischer Gestaltungshinweise bestritten werden, ihre Existenz angesichts des Mangels an wichtigen Unterstützungswerkzeugen verdeutlicht aber die starke Ausrichtung der Styleguides auf ein einheitliches optisches Erscheinungsbild aus Marketinggründen. Nicht die „carefully crafted interfaces“, wie Whiteside sie ge-

fordert hat (vgl. [Whiteside 85]), sind das Hauptanliegen von Styleguides, sondern ihr Zweck ist es, das gefällige Erscheinungsbild von grafischen Benutzungsoberflächen auf rationelle Weise mit einem hohen Wiedererkennungswert zu versehen.

Leider bieten die Werkzeuge zur Oberflächengestaltung den Software-Entwicklern nicht jene Unterstützung, die sie in ihrer realen Arbeitssituation am meisten brauchen würden. Da Anwendungssysteme ständig wachsen, verändert, verbessert und ergänzt werden (viel häufiger als völlig neue entstehen), müssen auch immer wieder neue Fenster, zusätzliche Menüpunkte, weitere Dialogboxen u.ä. in bereits bestehende Anwendungen integriert werden. Sollen diese Ergänzungen nicht durch zufällige Abweichungen vom anwendungsspezifischen „Look and Feel“ auffallen, müßte es dem Software-Entwickler mit Hilfe von Werkzeugen möglich sein, die spezifischen Designregeln einer Anwendung abzufragen, noch verbleibende Freiheitsgrade (z.B. für notwendige Hervorhebungen) festzustellen und die entsprechenden Merkmale gebündelt (z.B. wie in einer Default-Einstellung) anwenden zu können. Das Fehlen dieser Unterstützung verweist einmal mehr auf die fehlende Arbeitsanalyse bezüglich des Software-Entwicklungsprozesses selbst.

6.6 Grundzüge eines methodischen Rahmenkonzepts zur Überprüfung von Software

Software-ergonomische Normen und Gestaltungsgrundsätze formulieren nicht nur Orientierungsprinzipien, die bereits während der Entwicklung eines Software-Produkts bestimmte ergonomische Qualitäten des späteren Produkts wahrscheinlich machen oder sicherstellen sollen, sondern sie enthalten prinzipiell bereits den Verweis auf jene Beurteilungsdimensionen, anhand derer das fertige Programm sich einer empirischen Überprüfung seiner Eignung als Arbeitsmittel unterziehen läßt (vgl. etwa schon Dzida et al. 1977).

Mit Eignung ist dabei - im ergonomischen Kontext - nicht die Funktionalität und -sicherheit des Produkts gemeint (sie wird in diesem Fall als bereits geprüft vorausgesetzt), sondern die Gewähr dafür, daß das angebotene Arbeitsmittel nicht gegen die allgemein anerkannten Prinzipien einer menschengerechten Arbeitsgestaltung verstößt. Dies ist u.E. zugleich eine wesentlich präzisere Umschreibung des früher vielfach verwendeten Begriffs der ‚Benutzerfreundlichkeit‘.

Die Beurteilung der Eignung von Software im hier beschriebenen Sinne erfordert eine Operationalisierung der Gestaltungsprinzipien, d.h. ihre Überführung in Prüf- und Meßvorschriften, die im Zusammenhang mit einem verbindlichen methodischen Rahmenkonzept eine empirisch begründete und nachprüfbar Aussage darüber ermöglichen, unter welchen Voraussetzungen und mit welcher Sicherheit (bzw. Wahrscheinlichkeit) die Eignungsaussage - die letzten Endes immer eine *Prognose* bleibt - gerechtfertigt erscheint.

Obwohl verschiedentlich noch argumentiert wird, die bisher vorliegenden oder im Entwurfsstadium befindlichen software-ergonomischen Normen (und andere Richtlinien) seien einer entsprechenden Überprüfung nicht

zugänglich (vgl. [Wilke 91]), kann gezeigt werden, daß derartige Einwände als voreilig zurückzuweisen sind.

Sowohl im Rahmen der Evaluationsforschung (vgl. etwa [Weiss 74], [Rossi und Freeman 89], [Ravden 89], [Wottawa 90]; siehe vor allem auch das 'Program Evaluation Kit' [Herman 87] als auch der psychologischen Eignungsdiagnostik (vgl. [Wiggins 73] als state of the art) wurden differenzierte methodische Konzeptionen entwickelt, die sich zur Übertragung auf den Bereich der software-ergonomischen Eignungsüberprüfung und Qualitätssicherung anbieten.

Im Rahmen der Evaluation kann man - einer von Scriven in ([Scriven 67]) eingeführten Terminologie folgend - zwischen 'formativer' und 'summativer' Beurteilung unterscheiden: Erstere bezieht sich auf die fortlaufende Beurteilung eines Entwicklungsprozesses hinsichtlich seiner schrittweisen Verwirklichung vorgegebener Zielkriterien. Diese Art der Evaluation könnte also Entwickler von Anfang an unterstützen und ist auf längere Sicht der beste Weg, eine software-ergonomische Qualitätssicherung im gesamten Software-Lebenszyklus zu implementieren. Die 'summative' Evaluation bezieht sich dagegen auf ein abgeschlossenes Projekt oder Produkt. Auf Software angewandt, bietet sie die Möglichkeit, einem Anwender oder Nutzer gegenüber nachzuweisen, daß das gelieferte Programm bestimmten - z.B. ergonomischen - Anforderungen entspricht. Diese Anforderungen können sich auf die ISO-Normen 9241 beziehen.

Der Hinweis auf die Übertragbarkeit der in der psychologischen Eignungsdiagnostik entwickelten Ansätze auf Fragen der software-ergonomischen Qualitätsprüfung verdeutlicht das erforderliche Vorgehen aus einer anderen - wenngleich den Evaluationsansatz ergänzenden - Sicht. Zugleich läßt sich die These vertreten: Soweit Eignungsdiagnostik möglich und praktikierbar erscheint, kann nicht ernsthaft behauptet werden, die ergonomische 'Eignung' von Software - die sich in dieser Analogie um die Mitarbeit in einer Organisation 'bewirbt' - sei letzten Endes einer auf empirische Methoden gestützten Überprüfung nicht zugänglich.

Die Grundstruktur des eignungsdiagnostischen Ansatzes verdeutlicht Abb. 6.4: Zwischen zwei verschiedenen, unterschiedlich komplexen Situationen soll mit Hilfe nachprüfbarer (und möglichst quantifizierender) Operationen eine Beziehung hergestellt werden, die es erlaubt, das Verhalten des 'Bewerbers' in der komplexeren Situation aufgrund des in einer einfacheren Situation erfaßbaren Verhaltensausschnitts möglichst zutreffend vorherzusagen. Dies ist nur möglich, wenn Kriterien festgelegt werden, (a) welches Verhalten vorausgesagt werden soll, und (b) anhand welcher in der einfacheren Situation erfaßbaren Merkmale diese Prognose Erfolg verspricht.

Die kreisförmige Darstellung betont, daß das auf Analysen und Prognosen basierende Verfahren u.U. in mehreren Durchgängen schrittweise verfeinert und immer wieder auf seine Gültigkeit hin überprüft werden muß, was ein mehrfaches Durchlaufen der angedeuteten Schleife mit sich bringt.

Dies gilt auch hinsichtlich des bisherigen Fehlens einer Prüfvorschrift z.B. für die Dialog-Normen der ISO 9241-10: Das geeignete Verfahren muß empirisch erarbeitet werden. Daher ist Rauterbergs Auffassung zuzustimmen:

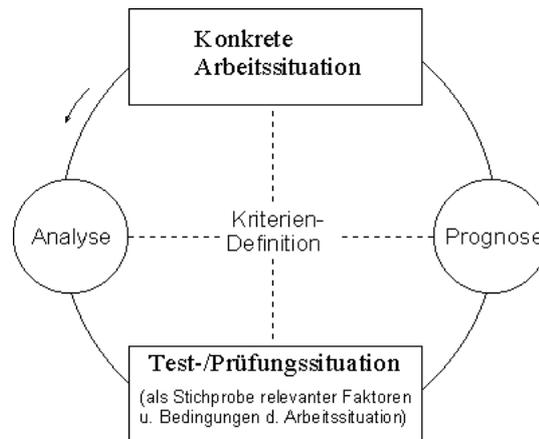


Abbildung 6.4: Prinzipieller Ablauf der Eignungsdiagnostik

„Jede Messung setzt eine Zuordnung von inhaltlichen Kriterien zu messbaren Eigenschaften der Realität voraus. Dazu stehen arbeitswissenschaftlich abgesicherte Kriterien, sowie das ausgereifte Methodenspektrum der Testtheorie, der Testplanung, der Skalierungstheorie und der angewandten Statistik zur Verfügung. Es ist zur Zeit also sehr gut möglich, die Gebrauchstauglichkeit von Software in einem hinreichenden Ausmaß messen und beurteilen zu können. Man muß es nur tun.“ (aus [Rauterberg 92] Seite 15)

Um dies zu tun, ist zunächst das allgemeine, zusammenfassende ergonomische Bewährungskriterium 'usability' (dem der 'Bewerber' Software genügen soll) in eine Reihe weiterer Komponenten aufzuspalten, die verschiedenen, voneinander unabhängigen Gesichtspunkten von und Zugangsweisen zu 'usability' (= Gebrauchstauglichkeit) entsprechen (vgl. Abb. 6.5).

Die erforderliche Dekomposition ermöglicht zunächst schon ISO 9241-11, indem definiert wird:

- „Gebrauchstauglichkeit: Das Ausmaß, in dem ein Produkt durch bestimmte Benutzer in einem bestimmten Nutzungskontext genutzt werden kann, um bestimmte Ziele effektiv, effizient und mit Zufriedenheit zu erreichen.
- Effektivität: Die Genauigkeit und Vollständigkeit, mit der Benutzer ein bestimmtes Ziel erreichen.
- Effizienz: Der im Verhältnis zur Genauigkeit und Vollständigkeit eingesetzte Aufwand, mit dem Benutzer ein bestimmtes Ziel erreichen.
- Zufriedenheit: Beeinträchtigungsfreiheit und Akzeptanz der Nutzung“ (zitiert aus EN 29241-11, 3.1-3.5).

Ein weiterer Schritt zur Dekomposition der Bewährungskriterien hat nunmehr u.E. bei der 'Effizienz' anzusetzen, indem der Begriff der 'Ressourcen' im

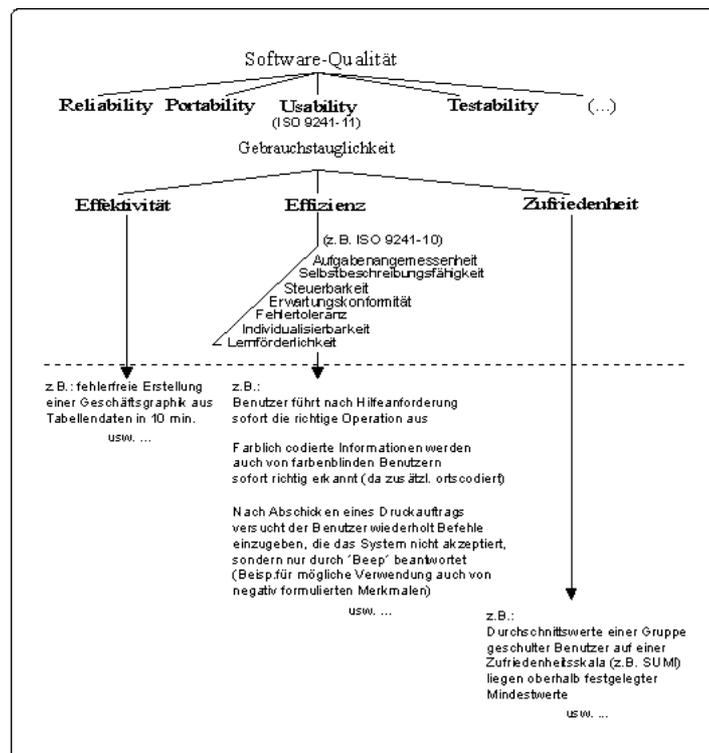


Abbildung 6.5: Dekomposition der Bewährungskriterien und Zuordnung von Prüfmerkmalen

Sinne der - gesundheitspolitisch relevanten - Belastungs-/Beanspruchungs- und Mental-Workload-Konzepte verstanden wird. Diesem Verständnis entsprechend bilden die Gestaltungsprinzipien der ISO-Normen 9241-10 und 9241-12 (und gegebenenfalls der Teile 13-18) weitere Subkomponenten der Effizienz, da sie auf jeweils unterschiedliche Weise - zumindest ihrer Intention nach, und im Sinne von Mindestanforderungen - der Beanspruchungsminde- rung dienen und die Verfügbarkeit der Ressourcen des Benutzers sichern können. Sollen Aufwand und Ertrag in ihrer wechselseitigen Beziehung betrachtet werden, so darf dieser Aspekt des Ressourcenverbrauchs seitens des Benutzers wohl kaum außer Betracht bleiben. In ISO CD 9241-11 wird diese - u.E. theoretisch gut begründbare - Interpretation von Effizienz zwar nicht herausgestellt; dies soll uns hier jedoch nicht daran hindern, eine andere Auffassung zu vertreten.

Auf der Basis der damit hinreichend detaillierten Dekomposition der Bewährungskriterien besteht die Aufgabe nunmehr darin, diesen Bewährungskriterien entsprechende Merkmale des 'Bewerbers' Software zu definieren, die sich in objektivierbarer Weise unter festgelegten Prüfbedingungen erfassen lassen (Testkriterien). Bei ihrer Auswahl kann man sich von den in den Normen der ISO 9241 genannten 'typical applications of the principle' und den zugehörigen Beispielen leiten lassen. Die Zuordnung solcher Testkriterien zu den einzelnen software-ergonomischen Prinzipien ist zunächst hypothetisch, läßt sich aber im Rahmen einer sogenannten „Konstruktvalidierung“ empirisch überprüfen und verifizieren (vgl. bereits [Cronbach und Meehl 55]; dies gehört längst zum Standard psychologischer Methodologie.

Während das prinzipielle Vorgehen damit recht einfach beschreibbar erscheint, darf der erhebliche analytische und methodische Aufwand nicht unterschätzt werden, der sich hinter der ISO-Definition von Gebrauchstauglichkeit (s.o.) in der knappen Formulierung verbirgt: „... with which specified users achieve specified goals in particular environments“, und damit eine Berücksichtigung der Spezifik des Nutzungskontexts erfordert.

Die einzelnen Komponenten des Prüfverfahrens zur 'Eignung' von Software lassen sich damit wie in Abb. 6.6 darstellen. Die Abbildung zeigt u.a., daß in der Regel die Eignungsprüfung nur dann den methodischen Anforderungen genügen kann, wenn in ihr repräsentativ alle für die reale Nutzungssituation charakteristischen Bedingungen enthalten sind. Die Interpretation des Prüfergebnisses ist - auf der Grundlage des varianzanalytischen Ansatzes (vgl. etwa [Bortz 89]) - nur möglich, wenn sich im einzelnen nachweisen läßt, welchen Einfluß die *unabhängigen* Variablen (Aufgaben, Benutzer und Rahmenbedingungen; Produktmerkmale) auf die in den Prüfergebnissen repräsentierten *abhängigen* Variablen, d.h. auf die ermittelten Effektivitäts-, Effizienz- und Zufriedenheitsmessungen haben (und welche Wechselwirkungen u.U. bestehen).

Dies erfordert - je nach Fragestellung - einen besonders hohen Aufwand bei der Untersuchungsplanung, der sich dann allerdings in Form wirklich aussagefähiger Resultate bezahlt macht.

Die skizzierten Anforderungen an das Untersuchungsdesign gelten auch - und in verschärftem Maße - für das Gebiet experimenteller software-

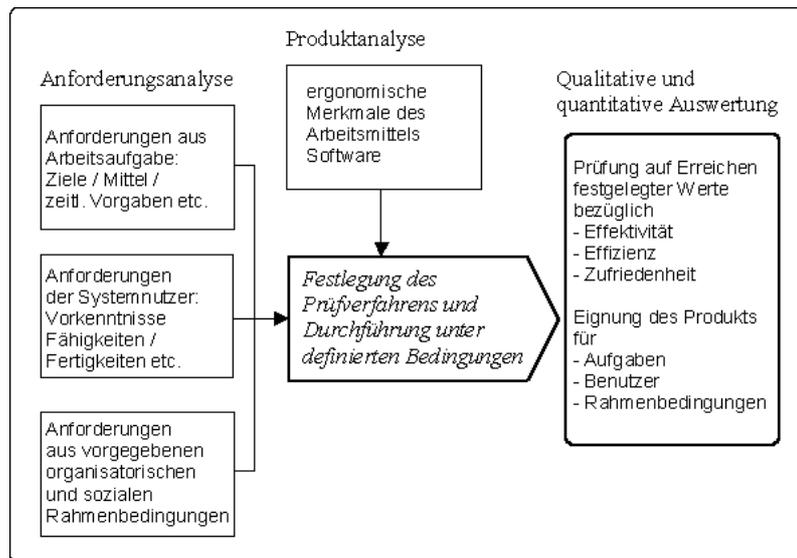


Abbildung 6.6: Hauptkomponenten der Eignungsprüfung für Software

ergonomischer Grundlagenforschung (das eigentlich bis heute kaum existiert). Die vorliegende Literatur zeigt, daß den bisherigen Ergebnissen empirisch-experimenteller Untersuchungen auf diesem Gebiet mit erheblicher Skepsis zu begegnen ist, und daß ihnen allenfalls Hinweisharakter (z.B. für die Formulierung von Arbeitshypothesen) zukommt. Dies liegt hauptsächlich daran, daß vielfach weder die im Experiment verwendeten Aufgaben, noch die als Benutzer fungierenden Versuchspersonen in hinreichendem Maße Repräsentanten vorfindbarer Nutzungskontexte sind.

Aufgrund solcher Repräsentanzmängel bisher vorliegender empirisch-experimenteller software-ergonomischer Arbeiten erscheint es durchaus verständlich, daß der Aufwand für realitätsnahe Eignungsüberprüfungen von Software als erheblich eingeschätzt wird, da nicht selektiv nur auf jene Fragen hin getestet werden kann, die im jeweiligen Fall noch als klärungsbedürftig erscheinen, während ansonsten der Hinweis auf die einschlägigen Grundlagenuntersuchungen genügen könnte.

Auf einige Möglichkeiten, den erforderlichen Prüfaufwand zu reduzieren, hat aber u.a. Dzida ([Dzida 91] S. 66) bereits hingewiesen. So gibt es (a) Fälle, in denen schon eine ausschließlich auf das Produkt bezogene Überprüfung genügt. Es läßt sich z.B. feststellen, ob überhaupt durchgängig eine Hilfefunktion zur Verfügung steht, und ob sie als kontextsensitiv gelten kann. In anderen Fällen (b) sind neben den Vorschriften der Norm zwar die Arbeitsaufgaben, nicht aber auch schon die realen Benutzer zu berücksichtigen; so etwa, wenn gefordert wird, der letzte Dialogschritt sollte, soweit seine Folgen reversibel sind und die Arbeitsaufgabe es zuläßt, zurückgenommen werden können. In vielen anderen Fällen verlangt eine den Normen entsprechende Eignungsüberprüfung allerdings tatsächlich die Untersuchung der unter realen Verhältnissen vorfindbaren komplexen Interaktion zwischen fachkompe-

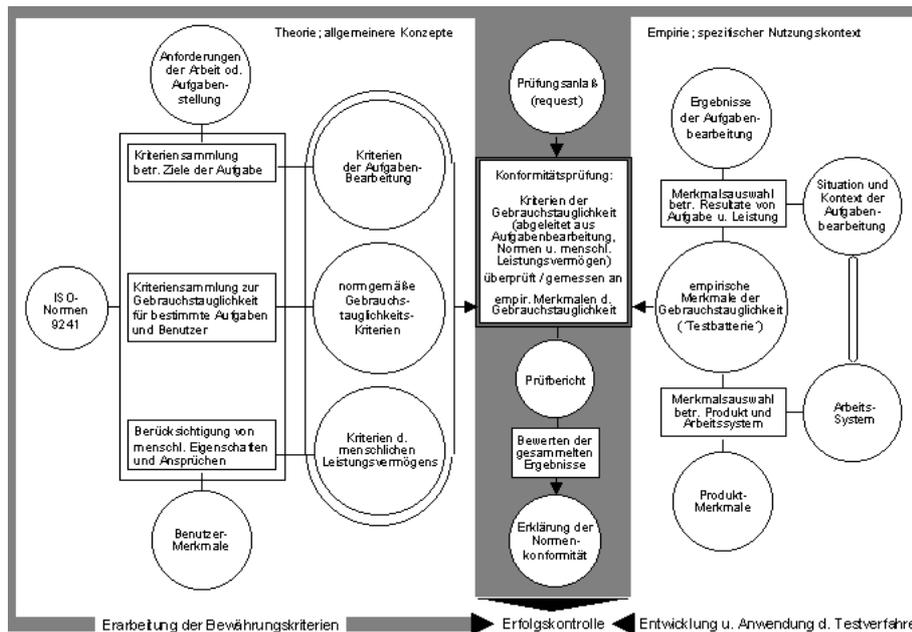


Abbildung 6.7: Modell zur software-ergonomischen Eignungsprüfung (angelehnt an [Dzida 94])

tenten Benutzern, ihren Arbeitsaufgaben und den in einem Unternehmen gegebenen Rahmenbedingungen.

Das gesamte, zur ergonomischen Eignungsprüfung bzw. Evaluation von Software anhand der ISO- bzw. CE-Normen erforderliche Vorgehen läßt sich zusammenfassend in einem Modell (Abb. 6.7) darstellen, das sich eng an einen von Dzida (1990) in die Normenentwicklung eingebrachten Entwurf anlehnt (vgl. [Dzida 93] und [Dzida 94])

Wir gehen davon aus, daß dieses Modell aufgrund der vorangehenden Erläuterungen hier nur noch einer knappen Erklärung bedarf.

Im linken Teil dieses Modells sind die Quellen angegeben, auf die bei der Zusammenstellung der Bewährungskriterien zurückzugreifen ist: Sie ergeben sich aus den Arbeitsaufgaben und den daraus abzuleitenden objektiven Anforderungen; aus Benutzermerkmalen, die als Eigenschaften (z.B. Charakteristika der menschlichen Informationsverarbeitung) und Ansprüche (z.B. Motive und Wertvorstellungen) in ihrem Zusammenwirken die Besonderheiten des menschlichen Leistungsvermögens bestimmen; sowie aus den Normen z.B. der ISO 9241 (bzw. CEN 29 241).

Im rechten Teil der Abbildung wird die Zusammenstellung all jener Merkmale - unter Angabe des geforderten Ausprägungsgrads - veranschaulicht, die im Test als Repräsentanten bzw. Prädiktoren der Bewährungskriterien (d.h. zusammenfassend: der software-ergonomischen 'Eignung') erfaßt und gemessen werden sollen. Hier geht es also um die Zusammenstellung der eigentlichen Testbatterie. Dabei sind insbesondere auch die Regelungen von ISO 9241-11 von Belang (und praxisnahe Beispiele in deren Anhängen zu finden).

Der mittlere Teil von Abb. 6.7 veranschaulicht die in einem Bericht zu dokumentierende Prüfung. Den Überlegungen von [Dzida 93] entsprechend wird in der Regel davon auszugehen sein, daß eine solche Prüfung eines besonderen Anlasses bedarf (z.B. Wunsch eines Anwenders) und sich - auch aus Kostengründen - zumeist selektiv auf jene Merkmale eines Softwareprodukts beschränken kann, die beispielsweise von bereits bewährten ergonomischen Lösungen (Mustern, reference materials) abweichen, oder die aufgrund einer veränderten Arbeitssituation (veränderter Aufgabenerfordernisse) einer Anpassung unterzogen wurden.

Da es bisher keine Erfahrungen mit der Anwendung eines solchen systematisch aufgebauten Evaluationskonzepts gibt, bietet die EG-Richtlinie 90/270/EWG nunmehr eine Chance, diese unbefriedigende Situation zu ändern. Der Hinweis auf das Fehlen bereits bewährter Prüf- und Meßverfahren sollte jedenfalls keine Rechtfertigung dafür sein, pauschal auf jegliche Schritte in diese Richtung zu verzichten. Auch für andere Gebiete der Ergonomie gilt schließlich, daß die heute verfügbaren Verfahren das Ergebnis einer langen Entwicklung und einer fortlaufenden kritischen Überprüfung der eingesetzten Verfahren sind.

6.7 Prüfverfahren - eine Kurzübersicht

Je nach Fragestellung, bereits vorliegenden Erkenntnissen und spezifischem Nutzungskontext sind verschiedenartige, qualitative oder quantitative Untersuchungsverfahren einzusetzen, die als Einzelkomponenten einer Umsetzung des hier modellhaft beschriebenen methodischen Gesamtkonzepts aufzufassen sind. Generell kommen dafür in Betracht:

- Verfahren der objektiven oder subjektiven Tätigkeits- bzw. Aufgabenanalyse;
- Analysen der grammatikalischen Komplexität von Dialogsequenzen mit Hilfe mathematischer Modelle;
- Checklisten, insbesondere zu Feststellung des Vorhandenseins geforderter Produkteigenschaften und spezifischer (eventuell einschränkender) Nutzungsbedingungen;
- Einstufungsverfahren (Skalen) für sog. Expertenratings zur Beurteilung der Benutzungsschnittstelle;
- skalierte, (teil-)standardisierte Fragebögen zur Erfassung von Benutzerurteilen (insbesondere zur Zufriedenheit);
- Verfahren zur Erfassung des Verhaltens repräsentativer Benutzergruppen bei der Bearbeitung repräsentativer Aufgaben (z.B. Videoaufzeichnungen, Logfiles, lautes Denken; u.U. auch deren Kombination) sowie
- hierauf bezogenen Zeit-, Leistungs- und Fehleranalysen;

- psychophysiologische Verfahren zur Erfassung mentaler Beanspruchungsprozesse (nur in wissenschaftlichen Untersuchungen als Ergänzung anderer Verfahren).

Einige bevorzugt einzusetzende Verfahren wurden bereits in Abschnitt 3.2.5 (Seite 39ff) näher erläutert.

Literaturverzeichnis

- [Ballstaedt 97] Ballstaedt, S.-P.: Wissensvermittlung. Die Gestaltung von Lernmaterial. Weinheim: Psychologie Verlags Union, 1997
- [Balzert 87] Balzert, H.: Gestaltungsziele der Software-Ergonomie. In Schönplüg & Wittstock 1987, 477-488
- [Balzert 88] Balzert, H. et al. (Eds.): Einführung in die Software-Ergonomie. Berlin: de Gruyter, 1988
- [Beimel 93] Beimel, J., Schindler, R. & Wandke, H.: Wie Experten der Software-Ergonomie den Teil 10 (Dialogue Principles) der ISO 9241 bewerten. In Rödiger 1993, 133-144
- [Bonitz 88] Bonitz, D. et al.: Zur Veränderung von Tätigkeitsstrukturen und Arbeitsinhalten durch den Einsatz von Rechnern. Zeitschrift für Arbeitswissenschaft 1988, 42 (14NF), 211-221
- [Bortz 89] Bortz, J.: Statistik. Berlin: Springer, 1989
- [Bullinger 93] Bullinger, H.-J.: Benutzergerechte Gestaltung von Software - eine Herausforderung an den Industriestandort Bundesrepublik Deutschland. In Coy et al. 1993, 17-38
- [Burmester et al. 97] Burmester, M. et al.: Das SANUS-Handbuch Bildschirmarbeit Eu-konform. Schriftenreihe der Bundesanstalt für Arbeitsschutz und Arbeitsmedizin – Forschung – FB 760. Bremerhaven: Wirtschaftsverlag NW, 1997
- [Çakir und Dzida 97] Çakir, A.; Dzida, W.: International ergonomic HCI standards. In: Helander, M. et al. (Eds.), Handbook of human-computer interaction, 2nd edition, 407-420. Amsterdam: Elsevier, 1997
- [Cronbach und Meehl 55] Cronbach, L.J.; Meehl, P.E.: Construct validity in psychological tests. Psychological Bulletin 1955, 52, 281-302
- [DoD 83] U.S. Dep. of Defense: Military standard: Human engineering design criteria for military systems, equipment and facilities. Washington, D.C.: U.S. Government printing office, 1983

- [Doherty 98] Doherty, P.B.: Learner control in asynchronous learning environments. ALN-Magazine 1998, Vol.2, Issue 2. Siehe http://www.aln.org/alnweb/magazine/vol2_issue2/doherty.htm
- [Duffy et al. 93] Duffy, T.M. et al. (Eds.): Designing environments for constructive learning. Berlin: Springer, 1993
- [Dutke 94] Dutke, S.: Mentale Modelle: Konstrukte des Wissens und Verstehens. Kognitionspsychologische Grundlagen für die Software-Ergonomie. Göttingen: Verlag für Angewandte Psychologie, 1994
- [Dzida 83] Dzida, W.: Das IFIP-Modell für Benutzerschnittstellen. Office Management 31, Sonderheft, 1983, 6-8
- [Dzida 85] Dzida, W.: Ergonomische Normen für die Dialoggestaltung. In Bullinger 1985, 430-444
- [Dzida 91] Dzida, W.: Software-Ergonomische Qualitätsprüfung — Utopie oder realistisches Ziel? In Çakir, A und Çakir (Eds.) Europa 1992 — Was bringen die Europäischen Regelwerke für Bildschirm-Arbeitsplätze. Tagungsband. Berlin: Ergonomic Institute 1991, 61-76
- [Dzida 93] Dzida, W.; Wiethoff, M.; Arnold, A.G.: ERGOguide. The quality assurance guide to ergonomic software. Delft / Sankt Augustin: Univ. of Technology / GMD, 1993
- [Dzida 94] Dzida, W.: Qualitätssicherung durch software-ergonomische Normen: In Eberleh, E; Oberquelle, H. und Oppermann, R. (Eds.): Einführung in die Software-Ergonomie. Berlin. de Gruyter 1994, 373-406
- [Eberleh 94] Eberleh, E.: Industrielle Gestaltungsrichtlinien für graphische Benutzungsoberflächen. In Eberleh, Oberquelle & Oppermann 1994, 145-195
- [Egan 88] Egan, D.E.: Individual Differences in Human-Computer Interaction. In Helander, M. (Eds.): Handbook of Human-Computer Interaction. Amsterdam: North-Holland, 1988
- [Eilers, Nachreiner und Böning 90] Eilers, K., Nachreiner, F. & Böning, E.: Zur subjektiven Skalierung psychischer Beanspruchung. Zeitschrift für Arbeitswissenschaft 1989, 43 (15NF), 217-223 (Teil 1) und 1990, 44 (16NF), 24-29 (Teil 2)
- [Friedrich 93] Friedrich, J. et al.: Die Gestaltung computergestützter Gruppenarbeit unter Berücksichtigung arbeitswissenschaftlicher Kriterien. Wirtschaftsinformatik 1993, 2, 101-110

- [Geis und Hartwig 98] Geis, T; Hartwig, R: Auf die Finger geschaut - Neue ISO-Norm für benutzergerechte interaktive Systeme; c't 14/98; Heise Verlag; Hannover; S. 168-172; 1998
- [Gerstenmaier und Mandl 95] Gerstenmaier, J. & Mandl, H.: Wissenserwerb unter konstruktivistischer Perspektive. Zeitschrift für Pädagogik 1995, 41, 867-888
- [Hacker und Richter 80] Hacker,W. & Richter,P.: Psychische Fehlbeanspruchung - Psychische Ermüdung, Monotonie, Sättigung und Streß. Berlin: DVW, 1980 (2.Aufl. 1984)
- [Hahn 95] Hahn, H. et al.: Arbeitssystem Bildschirmarbeit. Schriftenreihe der Bundesanstalt für Arbeitsschutz – Forschungsanwendung - Fa 31. Bremerhaven: Wirtschaftsverlag NW, 1995
- [Harasim 95] Harasim, L. et al.: Learning networks. A field guide to teaching and learning online. Cambridge, Ma.: MIT Press, 1995
- [Harker und Eason 92] Harker,S. & Eason,K.: Human Factors in the I.T. Software Design Process - The Way forward. In Galer,M.; Harker,S. & Ziegler,J. (Eds.): Methods and Tools in User-Centred Design for Information Technology. Amsterdam: North-Holland, 1992, 415-426
- [Hartwig 97] Hartwig, R: Gestaltung und Bewertung der Gebrauchstauglichkeit von Software - Überprüfung der Machbarkeit am Beispiel der Benutzungsoberfläche eines Moduls eines Schichtplangestaltungssystems (Diplomarbeit); Universität Oldenburg - Fachbereich Informatik, 1997
- [Hartwig, Triebe & Herczeg 2002] Hartwig, R.; Triebe, J.K.; Herczeg, M.: Styleguide — Richtlinien zur Qualitätssicherung bei der Realisierung von Studienmodulen im Projekt VFH. Medizinische Universität zu Lübeck — Institut für Multimediale und Interaktive Systeme, 2002
- [Herman 87] Herman,J.L. et al. : Evaluator's handbook. Newbury Park: Sage, 1987
- [Herczeg 94] Herczeg, M.: Software-Ergonomie. Grundlagen der Mensch-Computer-Kommunikation. Bonn etc.: Addison-Wesley und Oldenbourg-Verlag, 1994
- [ISO 6385] ISO 6385 / ENV 26385: Ergonomische Prinzipien bei der Konzeption von Arbeitssystemen; Berlin: Beuth Verlag
- [ISO 9241] ISO 9241 / EN 29241: Ergonomische Anforderungen für Bürotätigkeiten mit Bildschirmgeräten. Normenreihe, Teile 1 - 17; Berlin: Beuth Verlag

- [ISO 13407] ISO 13407: Benutzer- und aufgabengerechte Herstellungsprozesse, Berlin: Beuth Verlag, 1998
- [Issing 99] Issing, L.J.: Multimedia-integrierte Aus- und Weiterbildung. Vortrag, Berlin-Adlershof 6.7.1999. Siehe <http://userpage.fu-berlin.de/~medienfo/hp/Forschung/Adlershof.html>
- [IT Management 99] Rathjen, G: Ergonomisch, praktisch, gut — aus Zeitschrift: IT-Management 08/1999, IT-Verlag, Höhenkirchen, 1999, S. 18-24
- [Kirakowski und Corbett 90] Kirakowski, J. & Corbett, M.: Effective methodology for the study of HCI. Amsterdam: North-Holland, 1990
- [Klein 99] Klein, B.: Online-Tutorial "Einführung in die Kognitive Psychologie", 1999. Siehe <http://www.incops.de>
- [Koch und Reiterer 91] Koch, M.; Reiterer, H.; Min Tjoa, A.: Software-Ergonomie. Berlin: Springer, 1991
- [Maaß 93] Maaß, S.: Software-Ergonomie. Benutzer- und aufgabenorientierte Systemgestaltung. Informatik-Spektrum 16, 1993, 191-205
- [Microsoft 95] Microsoft: The Windows Guidelines for software design. Microsoft Press, 1995
- [Nachreiner und Mesenholl 93] Nachreiner, F. & Mesenholl, E.: Defizite der Software-Ergonomie - Ergebnisse einer Bilanzierung vorliegender Forschungsergebnisse zur Arbeit an Bildschirmgeräten. In Coy et al. 1993, 602-615
- [Naffah 91] Naffah, N.: A vision of the future. In Commission of the European Communities (Ed.): Market, competitors, development, future: Dimensions of change and strategic developments (ESPRIT Conference Brussels 27. November 1991). Brüssel: ECSC-EEC-EAEC, 1992
- [Nagl et al. 99] Nagl, M. et al.: Softwaretechnische Anforderungen an multimediale Lehr- und Lernsysteme. Studie der Forschergruppe SofTec NRW, September 1999. Siehe http://www.uni-paderborn.de/fachbereich/AG/schaefer/ag_dt/SofTecNRW/EndVersion/Studie.html
- [NASA 87] NASA: Man-system integration standard. Houston: NASA Johnson Space Center, 1987

- [Nielsen 93] Jakob Nielsen: Usability Engineering — Boston; San Diego; New York; London; Sydney; Tokyo; Toronto: AP Professional 1993
- [Oberquelle 91] Oberquelle, H. (Ed.): Kooperative Arbeit und Computerunterstützung. Stand und Perspektiven. Göttingen: Verlag für Angewandte Psychologie, 1991
- [Oberquelle et al. 83] Oberquelle, H. et al.: A view of human-machine communication and co-operation. International Journal of Man-Machine Studies 19, 4, 1983, 309-333
- [Oppermann 96] Oppermann, R. et al.: Der ISO 9241-Evaluator. Ergonomie & Informatik 27, 1996, 13-17
- [Plath 76] Plath, H.-E.: Zur Indikation von Belastungswirkungen kognitiver Tätigkeiten bei unterschiedlicher Schwierigkeit der Aufgabenbewältigung. In Hacker, W. (Ed.): Psychische Regulation von Arbeitstätigkeiten, 222-237. Berlin: DVW, 1976
- [Prümper 97] Prümper, J.: Der Benutzungsfragebogen ISONORM 9241/10: Ergebnisse zur Reliabilität und Validität. In: Lisowsky, R. et al. (Eds.), Software-Ergonomie '97, 253-262. Stuttgart: Teubner, 1997
- [Rauterberg 92] Rauterberg, M.: Messung der Gebrauchstauglichkeit interaktiver Software. In Görke, W. et al. (Eds.): Information als Produktionsfaktor, 211-221. Springer: Berlin, 1992
- [Rauterberg und Strohm 92] Rauterberg, M. & Strohm, O.: Work Organisation and Software Development. In Elzer, P & Haase, V.: Experience with the Management of Software Projects 1992. Oxford: Pergamon, 1992, 121-128.
- [Rauterberg 94] Rauterberg, M. et al.: Benutzerorientierte Software-Entwicklung - Konzepte, Methoden und Vorgehen zur Benutzerbeteiligung. Stuttgart: Teubner, 1994
- [Ravden 89] Ravden, S.J. & Johnson, G.I.: Evaluating usability of human-computer interfaces: A practical method. Chichester: Ellis Horwood, 1989
- [Rayner 83] Rayner, K. (Ed.): Eye movements in reading - Perceptual and language processes. New York: Academic, 1983
- [Rohr und Tauber 84] Rohr, G. & Tauber, M.J.: Representational frameworks and models for human-computer interfaces. In van der Veer, G.C. et al. (Eds.) Readings on cognitive ergonomics - mind and computers, 8-25. Berlin: Springer, 1984

- [Rossi und Freeman 89] Rossi,P.H. & Freeman,H.E.: Evaluation - A systematic approach (4th Edition). Newbury Park: Sage, 1989
- [Rouse 80] Rouse,W.B.: System engineering models of human-machine interaction. Amsterdam: North-Holland, 1980
- [Schönpflug 87] Schönpflug,W.: Beanspruchung und Belastung bei der Arbeit - Konzepte und Theorien. In Kleinbeck,U. Rutenfranz,J. (Eds.): Arbeitspsychologie. Enzyklopädie der Psychologie D/III/1, 130-184. Göttingen: Hogrefe, 1987
- [Scriven 67] Scriven,M.: The methodology of evaluation. In Tyler,R.W., Gagné,R.M. & Scriven,M. (Eds.): Perspectives of curriculum evaluation, 39-83. Chicago: Rand McNally, 1967
- [Smith und Mosier 84] Smith,S.L. & Mosier,J.N.: Guidelines for designing user interface software. Bedford: MITRE Corp., 1984
- [Streitz 85] Streitz,N.A.: Die Rolle von mentalen und konzeptuellen Modellen in der Mensch-Computer-Interaktion: Konsequenzen für die Software-Ergonomie? In Bullinger 1985, 280-292
- [Streitz 88] Streitz, N.A.: Fragestellungen und Forschungsstrategien der Software-Ergonomie. In: Balzert et al. 1988, 3-24
- [Souza und Bevan 90] Souza,F.de & Bevan,N.: The use of guidelines in menu interface design. In Diaper et al. 1990
- [Tetzlaff und Schwartz 91] Tetzlaff,L. & Schwartz,D.R.: The use of guidelines in interface design. Proceedings of CHI '91. New York: ACM, 1991, 329-333
- [Triebe und Wittstock 96] Triebe, J.K., Wittstock, M.: Anforderungskatalog für Softwareentwicklung und Anwendung. Bremerhaven: Wirtschaftsverlag NW, 1996
- [Triebe, Wittstock und Schiele 87] Triebe, J.K., Wittstock, M. & Schiele, F.: Arbeitswissenschaftliche Grundlagen der Software-Ergonomie. (Ed. Bundesanstalt für Arbeitsschutz, Sonder-schrift S 24). Bremerhaven: Wirtschaftsverlag NW, 1987
- [Triebe & Wittstock 98] Triebe, J.K.; Wittstock, M.:Anforderungen aus der Sicht von Sicherheit und Gesundheitsschutz an die Softwareentwicklung – Arbeitswissenschaftliche Erkenntnisse Nr. 107 — Bundesanstalt für Arbeitsschutz und Arbeitsmedizin; Dortmund; 1998
- [Triebe 2001] Triebe, J.K.: Fragebogen zur ergonomischen Evaluation von VFH-Lernmodulen durch Studierende. Universität Lübeck — Institut für Multimediale und Interaktive Systeme, 2001

- [Ulich 91] Ulich,E.: Arbeitspsychologie. Zürich / Stuttgart: Verlag d. Fachvereine / C.E.Poeschel, 1991 (2.Aufl. 1992)
- [Ulich et al. 91] Ulich,E. et al.: Task orientation and user-oriented dialogue design. Internat. Journal of Human Computer Interaction 1991, 3, 117-144
- [Volpert 87] Volpert,W.: Psychische Regulation von Arbeitstätigkeiten. In Kleinbeck,U. & Rutenfranz,J. (Eds.) Arbeitspsychologie, 1-41. Göttingen: Hogrefe, 1987
- [Volpert 93] Volpert,W.: Von der Software-Ergonomie zur Arbeitsinformatik. In Rödiger 1993, 51-65
- [VDI 88] VDI-Richtlinien Bürokommunikation (Entwürfe):
- VDI 5005: Software-Ergonomie in der Bürokommunikation. Düsseldorf, 1988
 - VDI 5010: Strategie und Vorgehen zur Einführung von Bürokommunikation. Düsseldorf, 1988
- [Weiss 74] Weiss,C.H.: Evaluation research. Englewood Cliffs: Prentice-Hall, 1972 (Deutsche Ausgabe: Evaluierungsforschung. Opladen: Westdeutscher Verlag, 1974)
- [Wessels 84] Wessels, M.G.: Kognitive Psychologie. New York: Harper & Row (UTB), 1984
- [Whiteside 85] Whiteside,J. et al.: User performance with command, menu, and iconic interfaces. Proceedings of CHI '85. New York: ACM, 1985, 185-191
- [Wieland-Eckelmann 92] Wieland-Eckelmann,R.: Kognition, Emotion und psychische Beanspruchung. Göttingen: Hogrefe, 1992
- [Wiley et al. 99] Wiley, D. et al.: Three common properties of efficient on-line instructional support systems. ALN Magazine, 1999, Vol. 3, Issue 2. Siehe: www.aln.org/alnweb/magazine/Vol3_issue2/wiley.htm
- [Wiggins 73] Wiggins,J.S.: Personality and prediction: Principles of personality assessment. Reading, Mass.: Addison-Wesley, 1973
- [Wilke 91] Wilke,K.-H.: Konformitätsprüfung für Produkte der Informationstechnik. In Çakir & Çakir 1991, 35-45
- [Wingert 83] Wingert,B.: Werkzeugenerfahrungen und Qualifikationsveränderungen beim CAD. Office Management, Sonderheft 1983, 31, 22-25

Literaturverzeichnis

- [Winn 93] Winn, W.: A constructivist critique of the assumptions of instructional design. In Duffy et al. 1993, 189-212
- [Wottawa 90] Wottawa, H. & H. Thierau: Lehrbuch Evaluation. Bern: Huber, 1990
- [Wydra 80] Wydra, F.T.: Learner-controlled instruction. Englewood Cliffs, NJ: Educ. Technology Publications, 1980

Abbildungsverzeichnis

| | | |
|------|--|-----|
| 2.1 | Aufgabenschwerpunkte und Ziele der Ergonomie | 10 |
| 2.2 | Software-Ergonomie als multi-disziplinäres Arbeitsgebiet (nach [Herczeg 94]) | 10 |
| 2.3 | Lernen im Zusammenwirken psychischer Prozesse | 17 |
| 2.4 | Gestaltungsschwerpunkte des soziotechnischen Systems „Virtuelle Hochschule“ | 21 |
| 3.1 | Die verschiedenen Prozessphasen und die möglichen Rücksprünge zu vorherigen Phasen | 24 |
| 4.1 | Unterschiedliche Vorstellungen eines Handlungsablaufes führen zur Belastung des Benutzers. | 44 |
| 4.2 | Gute Steuerbarkeit beim Web-Browsing (Netscape 4.6) | 47 |
| 4.3 | Unkonstruktive Fehlermeldung | 48 |
| 4.4 | Erwartungskonformer Dialog, der alle notwendigen Eingriffsmöglichkeiten und Informationen bietet. | 49 |
| 4.5 | Der Benutzer kann hier zum Beispiel die verwendeten Schriftarten selbst wählen, um sie seinen Sehgewohnheiten anzupassen | 51 |
| 4.6 | Lernförderliche Baumstruktur | 52 |
| 4.7 | Beispiel einer Webseite, die eigentlich alle Darstellungsprinzipien zugleich verletzt | 55 |
| 4.8 | Unklar gestalteter Dialog | 56 |
| 4.9 | Aufgeteilter und verbesserter Dialog (I) | 57 |
| 4.10 | Aufgeteilter und verbesserter Dialog (II) | 57 |
| 4.11 | Überladene Web-Seite | 59 |
| 4.12 | Schlechte Lesbarkeit wegen durchgehender Großschreibung | 62 |
| 4.13 | Verbesserte Lesbarkeit durch normale Groß-/Klein-Schreibung | 62 |
| 4.14 | Unklare Icons (Mac-Applikationen) | 63 |
| 6.1 | Systemkomponenten des Beanspruchungsprozesses (nach[Plath 76]) | 77 |
| 6.2 | Das erweiterte IFIP-Schnittstellenmodell | 85 |
| 6.3 | Die auf Software bezogenen ergonomischen Normen der ISO 9241 | 90 |
| 6.4 | Prinzipieller Ablauf der Eignungsdiagnostik | 105 |
| 6.5 | Dekomposition der Bewährungskriterien und Zuordnung von Prüfmerkmalen | 106 |
| 6.6 | Hauptkomponenten der Eignungsprüfung für Software | 108 |

| | | |
|-----|---|-----|
| 6.7 | Modell zur software-ergonomischen Eignungsprüfung (angelehnt an [Dzida 94]) | 109 |
|-----|---|-----|

Tabellenverzeichnis

| | | |
|-----|---|----|
| 3.1 | Die verschiedenen Rollen und Phasen bei der Erstellung einer Lerneinheit. | 28 |
| 3.2 | Beispiel für verschiedene Aufwandsarten und mögliche Arbeitsabschnitte bei der Nutzung des virtuellen Studienangebotes bezogen auf ein einzelnes Lernmodul | 38 |
| 6.1 | ISO Normenreihe ‚Ergonomic requirements for office work with visual display terminals (VDTs)‘ - ISO 9241 Part 1-18 (unter CE 29241 als europäische Norm übernommen). Abkürzungen zum Status bedeuten (bei zunehmender Vorläufigkeit des bis jetzt vorliegenden Textes): IS=International Standard, DIS=Draft International Standard, CD=Committee Draft, WD=Working Draft | 88 |