

Über die Quantifizierung softwareergonomischer Richtlinien

Matthias Rauterberg

Institut für Arbeitspsychologie, Eidgenössische Technische Hochschule
Nelkenstrasse 11, CH-8092 Zürich
rauterberg@ifap.bepr.ethz.ch

Zusammenfassung

Ausgehend von einer Analyse und Bewertung der bisher entwickelten Richtlinien- und Regelsätze für eine softwareergonomische Gestaltung von Benutzungsoberflächen wird die Notwendigkeit zur Ableitung möglichst quantitativer Metriken aufgezeigt. Es wird eine allgemeine Systemarchitektur für die verschiedenen systemtechnischen Komponenten von Benutzungsoberflächen vorgestellt. Für die Quantifizierung von softwareergonomischen Richtlinien wird zunächst ein allgemein anwendbares Beschreibungskonzept für Benutzungsoberflächen dargestellt. Mittels dieses Beschreibungskonzeptes können wichtige Eigenschaften von Benutzungsoberflächen definiert werden, welche die Ableitung von softwareergonomisch relevanten Metriken ermöglichen. Für die Richtlinie 'Feedback' werden zwei verschiedene quantitative Metriken hergeleitet und die zugehörigen Maße zur Beschreibung von drei verschiedenen interaktiven Systemen mit je zwei unterschiedlichen Oberflächen herangezogen. Für die Richtlinie 'Flexibilität' werden drei quantitative Metriken hergeleitet und die entsprechenden Maße ebenfalls zur Beschreibung der sechs verschiedenen Oberflächen verwendet. Die Validierung der Maße für Flexibilität erfolgte durch zwei eigene, sowie einen fremden Vergleichstest. Anhand der gefundenen Ergebnisse kann für die zwei Maße der interaktiven Flexibilität ein Schwell- bzw. Grenzwert angegeben werden. Erst wenn dieser Schwellwert überschritten wird, läßt sich auch ein Benutzungsvorteil aufgrund entsprechender Flexibilität empirisch nachweisen. Mittels weiterer Analysen von veröffentlichten Vergleichstests konnte dieser Grenzwert zusätzlich bestätigt werden. Abschliessend werden die gewonnenen Erkenntnisse im Kontext der bereits bekannten Ergebnisse softwareergonomischer Forschung diskutiert und Wege für zukünftige Forschungsgebiete aufgezeigt.

Stichworte: Metrik, Softwareergonomie, Test.

1. Einleitung

"Es ist derzeit noch nicht möglich, die Erfüllung einzelner ... Leitsätze objektiv zu überprüfen, da geeignete Überprüfungsverfahren noch nicht bekannt sind. Wenn Prüfverfahren bekannt sind, bedarf es noch einer

Weiterentwicklung dieser oder weiterer Normen, z.B. hinsichtlich quantifizierbarer Größen und anwendungsspezifischer Anforderungen." (DIN 66 234 [12])

Solange das Zitat aus der DIN 66 234 [12] seine Gültigkeit noch nicht vollständig verloren hat, gab und gibt es die verschiedensten Ansätze zur Entwicklung quantifizierbarer Metriken für softwareergonomische Aspekte. Bisher gab es mindestens drei europäische Projekte zur Lösung der aufgezeigten Problematik:

1. Das EVADIS-Projekt I und II bei der Gesellschaft für Mathematik und Datenverarbeitung [28].
2. Das PROTOS-Projekt an der Technischen Universität in München [26].
3. Das Esprit-Projekt MUSiC [35].

Für eine Einordnung der Ergebnisse dieser drei Projekte kann die Unterscheidung in die folgenden vier Methoden bzw. Meßansätze herangezogen werden (in Anlehnung an [40] S.74; [23] S.600; [37]; siehe ausführlicher [32]):

- Der *formalanalytische Meßansatz*; die Benutzungsfreundlichkeit des Systems wird in ergonomischen Eigenschaften eines formalen Modelles (z.B. GOMS, CCT; siehe [10]) bestimmt.
- Der *benutzerzentrierte Meßansatz*; subjektive Urteile der Benutzer bilden die Grundlage dieses Ansatzes (z.B. subjektive Ratings, psychomentele Beanspruchungen, etc. [32]).
- Der *interaktionszentrierte Meßansatz*; Benutzungsfreundlichkeit läßt sich über Eigenschaften der Interaktion zwischen Benutzer und System messen (z.B. Performanz, Anzahl Tastendrucke, etc. [32]).
- Der *produktzentrierte Meßansatz*; die Benutzungsfreundlichkeit des Systems wird in ergonomischen Eigenschaften des Produktes selbst bestimmt (z.B. Maskenaufbau, Interaktionstechnik, etc.). Zusätzlich kann man bei diesem Meßansatz zwischen 'subjektiven' und 'objektiven' Methoden unterscheiden: die Expertenevaluation als eher 'subjektive' Schätzmethode und die Berechnung quantitativer Kennwerte bzw. Metriken als objektive Meßmethode.

Die Ergebnisse des PROTOS- und des MUSiC-Projektes können am besten dem benutzer- und *interaktionszentrierten* Meßansatz, die Ergebnisse des EVADIS-Projektes eher der 'subjektiven' Evaluationsmethode des *produktzentrierten* Meßansatzes zugeordnet werden. In keinem der drei erwähnten Projekte werden objektive

Meßvorschriften abgeleitet und empirisch validiert, welche die softwareergonomische Güte direkt an dem Produkt selbst feststellen ließe. Diese Art der Produktgütebestimmung wurde der *produkt-zentrierte Meßansatz mittels objektiver Meßvorschriften* genannt. Englisch [15] entwickelte einen ersten Ansatz, die softwareergonomische Produktgüte von CAD-Systemen teilweise mittels eines objektiven Maßes für die Menükomplexität zu bestimmen. Leider mußte sich Englisch ([15] S. 167) – wahrscheinlich wegen der enormen Komplexität der analysierten Softwareprodukte – bei der 'Validierung' seiner Maße mit "hypothetischen" Kennwertabschätzungen begnügen.

2. Herleitung eines allgemeinen Beschreibungskonzeptes

Ausgehend von dem Anspruch, objektive Metriken bzw. Meßvorschriften zur Bestimmung der softwareergonomischen Produktgüte herzuleiten und empirisch zu validieren, wurde folgendermaßen vorgegangen.

(1) *Herleitung eines angemessenen Konzeptes zur abstrakten Beschreibung von Benutzungsoberflächen*: Der Kritik am Seeheim- bzw. IFIP-Modell folgend (siehe [8], [14], [7]) wird zwischen Dialog- und Anwendungskomponente unterschieden (Abbildung 1). Wenn eine Funktion das jeweilige Anwendungsobjekt (z.B. Datenbank, Textdokument, gesteuerter Prozeß etc.) in seinem Zustand bzgl. seines Inhaltes verändert, ist sie eine *Anwendungsfunktion* (z.B. 'Update', 'Format kursiv' etc.). Diesen Anwendungsfunktionen wird ein *verborgener anwendungsbezogener funktionaler Interaktionspunkt* (VAFIP) zugeordnet. Alle anderen Funktionen sind 'reine' *Dialogfunktionen*; diesen Dialogfunktionen wird ebenfalls ein *verborgener dialogbezogener funktionaler Interaktionspunkt* (VDFIP) zugeordnet. Diese Aufteilung in Dialog- und Anwendungsfunktionen entspricht weitgehend der klassischen Unterscheidung in *Form* und *Inhalt*.

Die Menge aller verborgenen Funktionen (VF) ist die Vereinigung der Menge aller verborgenen Anwendungsfunktionen (VAFIP) mit der Menge aller Dialogfunktionen (VDFIP). Die Menge der Anwendungsfunktionen sollte gemäß Ulich ([38] S. 482) *aufgabenangemessen* sein; die Menge der Dialogfunktionen sollte minimal und einfach zu benutzen sein. Durch diese Art der Aufteilung in Dialog- und Anwendungsfunktionen kann das von Streitz [36] beschriebene Sach- und Dialogproblem eindeutig unterschiedlichen Funktionsgruppen zugeordnet werden. Ebenso lassen sich die interaktiven Objekte in wahrnehmbare Dialogobjekte (WDO, z.B. das Papierkorb-Ikon einer Desktopoberfläche) und wahrnehmbare Anwendungsobjekte (WAO) unterscheiden (siehe Abbildung 2 und 6).

(2) *Erstellung eines möglichst einfachen, allgemein anwendbaren Beschreibungskonzeptes für handlungspsychologisch relevante Eigenschaften von Benutzungsoberflächen*: Es wird (a) gemäß Schritt 1 zwischen Dialog- und Anwendungsfunktionen, (b) zwischen dem Objekt- und Funktionsraum und (c) aus Sicht der Benutzer zwischen wahrnehmbaren und verborgenen Funktionen unterschieden. In Abbildung 2 ist eine vollständige Auflistung aller unterschiedenen Beschreibungsstrukturen gegeben. Der Funktionsraum (FR) besteht aus den wahrnehmbaren und den verborgenen *funktionalen Interaktionspunkten* (FIP abgekürzt) getrennt nach Anwendungs- und Dialogfunktionen (AFIP bzw. DFIP).

(3) Die systeminternen Algorithmen, welche während der Interaktion aktivierbar sind, werden *Funktionen* genannt; diese – primär aus Benutzersicht nicht wahrnehmbaren – Funktionen werden durch ihre *funktionalen Interaktionspunkte* (FIP) beschrieben. Die Aktivierung einer Funktion erfolgt aus Sicht der Benutzer durch eine *Operation*, welche die Eingabe eines *Operators* bedeutet (Abbildung 3). Die Eingabe eines Operators kann aus der Ausführung einer oder mehrerer *Aktionen* bestehen; z.B. die Eingabe eines Kommandos als Operator setzt sich aus den Aktionen der einzelnen Tastendrucke zusammen. Die Eingabe erfolgt hinter dem Kommando-Prompt (dem WF, siehe Abbildung 3).

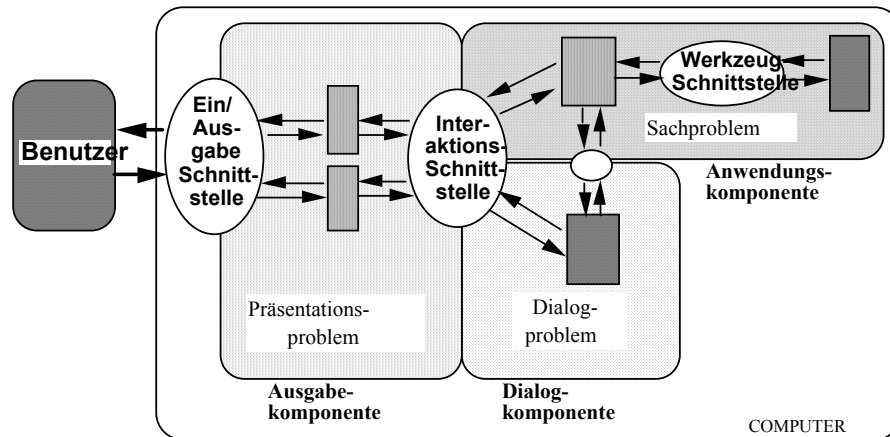


Abbildung 1: Übersicht über die verschiedenen Schnittstellen eines interaktiven Computersystems und die zugeordneten Problembereiche in Form einer Petri-Netzdarstellung (ohne Berücksichtigung der Organisationsschnittstelle; siehe [34]).

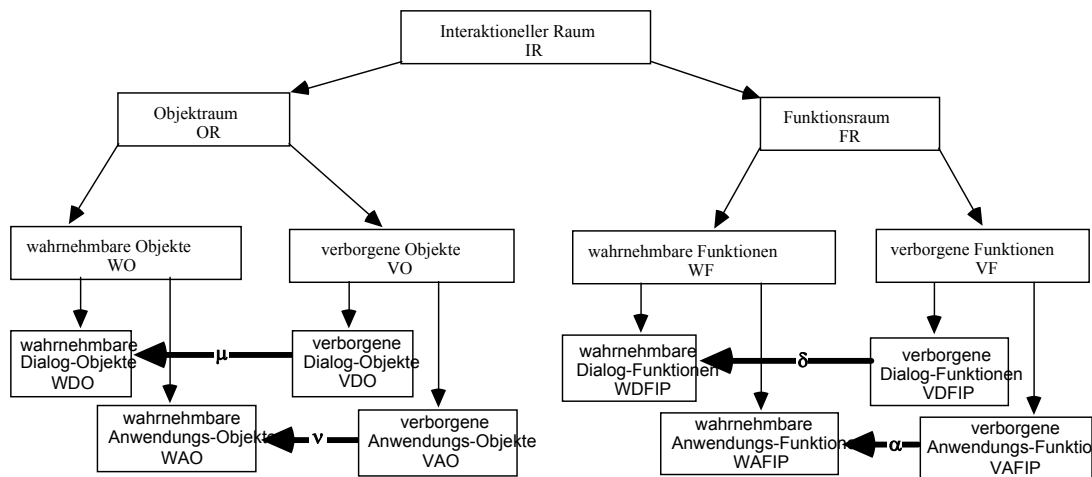


Abbildung 2: Der interaktionelle Raum (IR) besteht aus dem Objektraum (OR) und dem Funktionsraum (FR). OR und FR lassen sich weiter in wahrnehmbare (W*) und verborgene (V*) Bereiche bezüglich der Dialog- (*D*) und der Anwendungs-Komponente (*A*) aufteilen. Der FR wird durch funktionale Interaktionspunkte (*FIP) beschrieben.

Um die verborgenen Funktionen (VAFIPe bzw. VDFIPe) dem Benutzer auf der Ein/Ausgabeschnittstelle wahrnehmbar zu präsentieren, werden die Abbildungsvorschriften δ und α für die VDFIPe bzw. die VAFIPe, sowie die Abbildungsvorschriften μ und ν für die VDOe bzw. die VAOe eingeführt. Es ist zur Zeit ein noch weitgehend offenes Forschungsfeld, wie diese Abbildungsfunktionen auszusehen haben. Die Styleguides der Hersteller (SAA/CUA, OSF/Motif, etc.; z.B. [4] [21] [25]) legen lediglich die Repräsentationsformen der VDFIPe und VDOe fest. Ob es jedoch sinnvoll ist, z.B. für einen VAFIP ein Icon oder eine Menüoption zu wählen, bleibt zukünftiger Forschung vorbehalten.

Anhand der schematischen Darstellung in Abbildung 4 läßt sich aufzeigen, daß die verborgenen Anwendungsfunktionen (VAFIP-3, VAFIP-4 und VAFIP-5) nur einen gemeinsamen WAFIP besitzen; diese Situation ist typisch für Kommandooberflächen (siehe auch [31]). Ganz anders dagegen sieht es bei direktmanipulierbaren Oberflächen (z.B. Desktopoberflächen) aus. Hier gibt es oftmals zu jedem VAFIP bzw. VDFIP eine entsprechend wahrnehmbare Repräsentationsform (WAFIP bzw. WDFIP) auf der Ein/Ausgabeschnittstelle (siehe Abbildung 6). Wenn jedem VAFIP bzw. VDFIP mindestens ein WAFIP bzw. WDFIP zugeordnet ist, dann ist diese Oberfläche durch 100% funktionales Feedback gekenn-

zeichnet. Versteckte Funktionalität (z.B. über die 'modifier keys') bei graphischen Oberflächen wird oft nicht benutzt. Erst eine wahrnehmbare Repräsentation (visuell, auditiv, taktil) erlaubt es dem Benutzer, auch diese Funktionalität in seinen interaktiven Bearbeitungsprozeß einzubeziehen.

Ein Dialogkontext (D) läßt sich am besten über die aktuell vorhandene Menge an interaktiven Objekten (AO und DO), sowie der aktuell aktivierbaren Menge an Funktionen (AFIP und DFIP) charakterisieren. Jede Änderung bei einer dieser Mengen ergibt einen neuen Dialogkontext.

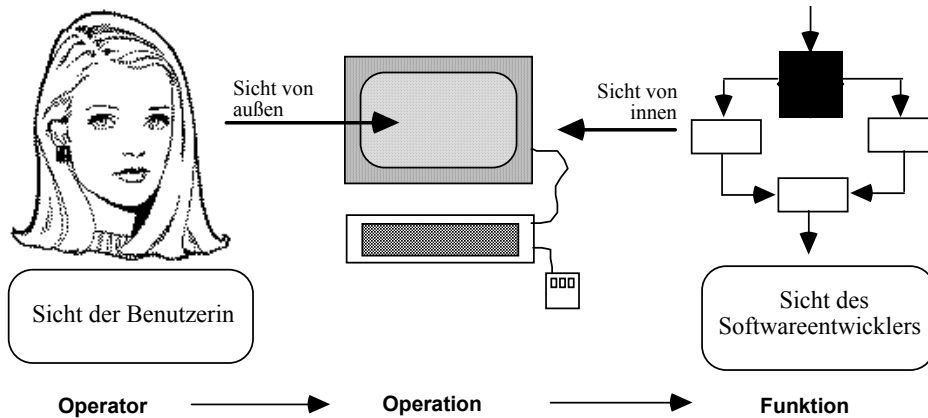


Abbildung 3: Die beiden unterschiedlichen Sichten auf die Ein/Ausgabe-Schnittstelle: (1) die Benutzersicht und (2) die Softwareentwicklersicht.

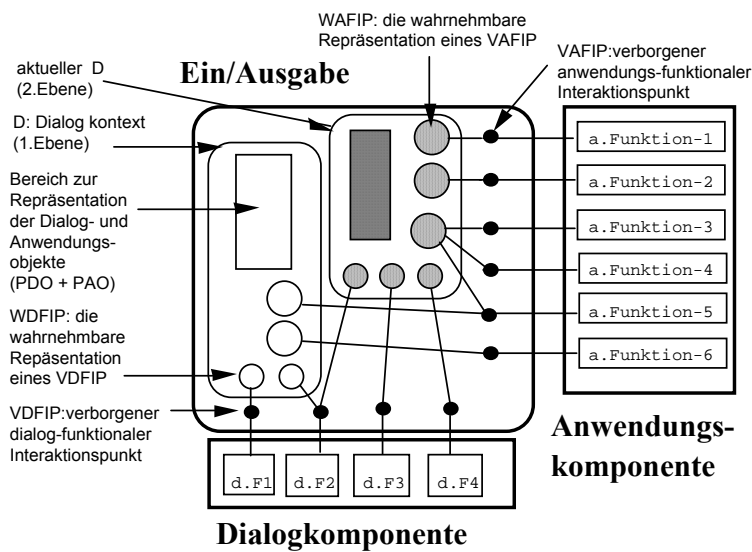


Abbildung 4: Schematische Darstellung der drei Komponenten eines interaktiven Systems mit den verschiedenen verborgenen und wahrnehmbaren Interaktionspunkten.

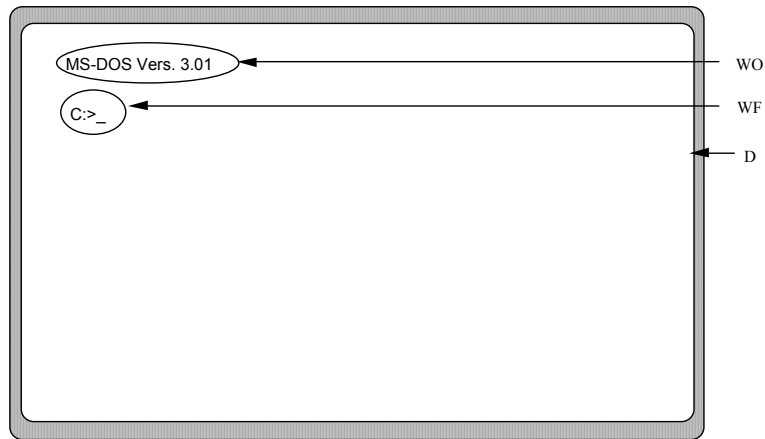


Abbildung 5: Ein möglicher Dialogkontext (D) der Kommandooberfläche des Betriebssystems MsDOS.

Ein Dialogkontext D ist ein Neun-Tupel der folgenden Form: $D = (VDFIP_m, WDFIP_n, VAFIP_t, WAFIP_s, DKZ_d, WDKZ_e, AKZ_a, WAKZ_b, OR_o)$; mit $VDFIP_m$ als die Menge mit m -Elementen aus der Menge aller verborgenen dialogfunktionalen Interaktionspunkte; mit $WDFIP_n$ als die Menge mit n -Elementen aus der Menge aller wahrnehmbaren dialogfunktionalen Interaktionspunkte gemäss der kontextspezifischen Abbildungsvorschrift δ der m $VDFIP_e$ auf die Menge aller möglichen Repräsentationsformen für Funktionen (RF; z.B. Menüoptionen, Icons etc.); mit $VAFIP_t$ als die Menge mit t -Elementen aus der Menge aller anwendungsfunktionalen Interaktionspunkte; mit $WAFIP_s$ als die Menge mit s -Elementen aus der Menge aller wahrnehmbaren anwendungsfunktionalen Interaktionspunkte gemäss der kontextspezifischen Abbildungs-

vorschrift α der t $VDFIP_e$ auf RF; mit DKZ_d als die Menge aller Zustände der Dialogkomponente (d an der Zahl); mit $WDKZ_e$ als die Menge aller e wahrnehmbaren Zustände der Dialogkomponente gemäss der kontextspezifischen Abbildungsfunktion ζ_δ der d Elemente von DKZ auf die Menge aller repräsentierbaren Zustände (RZ); mit AKZ_a als die Menge aller Zustände der Anwendungskomponente (a an der Zahl); mit $WAKZ_b$ als die Menge aller b wahrnehmbaren Zustände der Anwendungskomponente gemäß der kontextspezifischen Abbildungsfunktion ζ_α der a Elemente von AKZ auf die Menge aller repräsentierbaren Zustände RZ; sowie mit OR_o als die Menge aller interaktiven Objekte im Objekt-raum (o an der Zahl).

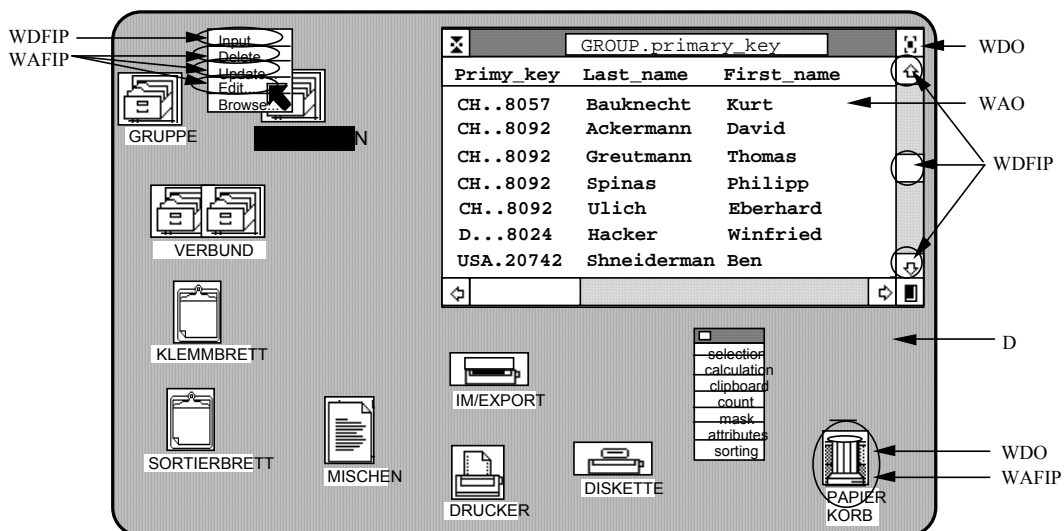


Abbildung 6: Ein möglicher Dialogkontext (D) der direktmanipulierbaren Oberfläche eines fiktiven Datenbanksystems mit beispielhaft markierten Interaktionspunkten (WAFIP & WDFIP) und den verschiedenen interaktiven Objekten (WAO & WDO).

3. Herleitung von zwei Metriken für 'Feedback'

Eine Handlung ist die kleinste psychologische Einheit einer willensmäßig gesteuerten Tätigkeit. Handlungen – insbesondere auch kognitive Problemlösehandlungen – verlaufen nach einem fünfstufigen, psychischen Regulationsprozeß jeweils auf den folgenden Ebenen [19]:

1. Vorsatzbildung, Planerstellung und Zielvorgabe;
2. Orientierung über die Zustandsbedingungen der Aufgabe, den Ausführungsbedingungen und den Handlungsbedingungen;
3. Entwerfen eines Aktionsprogrammes durch die Bildung eines Ergebnis- und Tätigkeitsmodelles;
4. Entscheidung über Ausführungsweisen und Ausführen des Entschlusses durch Anwendung eines entsprechenden Operators bzw. einer Operatorsequenz;
5. Kontrollieren des Ausführens über wahrgenommene Zustandsparameter.

Damit der Benutzer für die korrekte Handlungsplanung und -ausführung einen geeigneten Operator auswählen kann, hat es sich als vorteilhaft herausgestellt, wenn diesem Operator eine wahrnehmbare Repräsentationsform zugeordnet ist (siehe zum Thema *Transparenz* und *Feedback* bei [17], [39], [13]). Das Verhältnis von wahrnehmbaren Interaktionspunkten der Operatoren zu den verborgenen funktionalen Interaktionspunkten läßt sich daher als eine Metrik für Feedback pro Dialogkontext definieren (zum Begriff der Metrik siehe [5] S. 442). Wie die Repräsentation im konkreten Einzelfall (Ikon, Menüoption etc.) jeweils gestaltet ist, bleibt vorerst unberücksichtigt.

		K	
AFBF	=	$\sum_{d=1}^{K}$	$(\#WF_d / \#VF_d)$
AFBF	absolutes Maß für Feedback bzgl. der interaktiven Funktionen;		
$\#WF_d$	Anzahl an allen wahrnehmbaren funktionalen Interaktionspunkten bezüglich Dialogkontext d;		
$\#VF_d$	Anzahl an allen verborgenen funktionalen Interaktionspunkten bezüglich Dialogkontext d;		
K	Anzahl an analysierten Dialogkontexten D.		

Das so definierte Maß AFBF ist jedoch in seiner Größe noch abhängig von der absoluten Anzahl an analysierten Dialogkontexten und kann daher nur bedingt zum Vergleich zwischen verschiedenen Softwareprodukten her-

angezogen werden. Um diesen Nachteil auszugleichen, wird das relative Maß RFBF definiert. Es läßt sich das Maß RFBF aus dem zugehörigen Maß AFBF wie folgt herleiten: $RFBF = 100\% * AFBF / K$. Die Anwendung dieser beiden Maße auf jeden Dialogkontext der kompletten Interaktionsstruktur eines beliebigen Softwareproduktes erlaubt die einfache Berechnung der definierten Kennwerte pro Metrik. Als Problem hat es sich gezeigt, daß nicht immer alle verborgenen FIPE bekannt sind. Meistens muß man sich bei der Anwendung dieser Metriken auf die dokumentierte Funktionalität beschränken, obwohl oftmals zusätzliche Funktionen implementiert sind oder dokumentierte Funktionen – was eigentlich nicht vorkommen sollte – nicht ausführbar sind.

4. Herleitung von drei Metriken für 'Flexibilität'

Für die benutzungsgerechte Gestaltung der Dialogstruktur hat sich das Kriterium der 'Flexibilität' ([38] [13]) als besonders relevant herausgestellt. Je größer die Flexibilität eines gegebenen Handlungsfeldes ist, desto besser können individuelle Unterschiede im 'differenziell-dynamischen' Sinne [38] berücksichtigt werden. Jeder Dialogkontext kann als ein lokales Handlungsfeld betrachtet werden. Es gibt Operatoren, welche den Dialogkontext bewahren, und solche, welche einen Dialogkontextwechsel herbeiführen. Entsprechend kann man zwischen einer *globalen* und einer *lokalen* Flexibilität unterscheiden: (1) durchschnittliche Anzahl Operationen pro Dialogkontext (lokale Flexibilität), als auch (2) durchschnittliche Anzahl möglicher Folgedialogkontexte (globale Flexibilität). Der Wechsel eines Dialogkontextes ist durch die Pfeile in Abbildung 7 symbolisiert. Pro Dialogkontext kann dieser *interaktive Verzweigungsgrad (IVG)* als Maß für die *globale* Flexibilität definiert werden.

		K	
$IVG = 1/K$	\sum	$\sum_{d=1}^{K}$	$Post(D_{d,f})$
			$f \in VF$
IVG	Maß für den interaktiven Verzweigungsgrad (globale Flexibilität);		
IS	ein Interaktionsschritt ist eine Operation, welche zu einem Dialogkontextwechsel führen kann oder den aktuellen Dialogkontext erhält;		
lng	Länge des Interaktionspfades (Pfd), welche der Anzahl an ISen entspricht;		
$Post(D_{d,f})$	Erreichbarkeitsfunktion aller unmittelbar zu $D_{d,f}$ folgenden, unterschiedlichen Dia-		

logkontexte mit $\ln(\text{Pfd}[\text{IS}_f]) = 1$ für alle $f \in \text{VF}$; Post ist 1 für alle erreichbaren Dialogkontexte, 0 sonst.

K Anzahl der analysierten Dialogkontexte D.

Bei direktmanipulierbaren Oberflächen ist die Anzahl möglicher Dialogkontexte oftmals sehr groß. Um z.B. die Interaktionsstruktur graphischer Oberflächen (GUI) vollständig zu beschreiben, reichen daher kontextfreie Sprachen bzw. Zustandsübergangsdiagramme (ZÜD, z.B. Abbildung 7) nicht aus.

"Für realistische graphische Anwendungen ist eine Spezifikation mit ZÜD nicht durchführbar. Allerdings ist die Komplexitätsbeherrschung graphischer Interaktionen auch mit anderen existierenden Spezifi-

kationsmittel nicht gelöst, weil nicht die Spezifikationsmittel selbst, sondern der Umfang graphischer Benutzerschnittstellen und die Feinregulierung der Beschreibung die Problemursachen darstellen" ([20] S. 284).

Wenn man sich jedoch für typische Aufgaben auf alle mindestens hierzu notwendigen Dialogkontexte beschränkt, erhält man eine *untere Abschätzung* als Kennwert. Die im folgenden aufgeführten Werte von GUI-Oberflächen stellen somit *untere Grenzwerte* dar. Der hier gewählte pragmatische Ansatz ist insofern als konservativ anzusehen, als die GUI-Oberflächen im Einzelfall größere Kennwerte erhalten können. Für die sieben ausschnitthaft abgebildeten Dialogkontexte in Abbildung 7 beträgt z.B. $\text{IVG} = 2,14$.

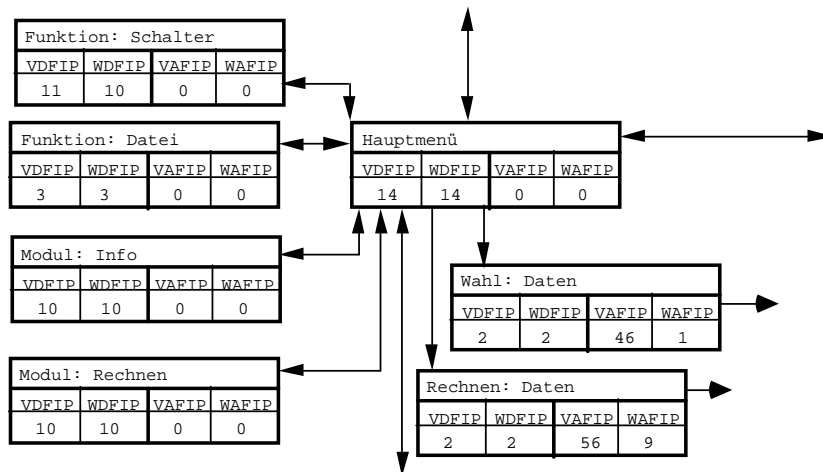


Abbildung 7: Ausschnitt aus einem Interaktionsstrukturschema als Zustandsübergangsdiagramm für die CUI-Oberfläche aus [30] und [33]. [Legende: VDFIP = verborgener dialogfunktionaler Interaktionspunkt; WDFIP = wahrnehmbarer dialogfunktionaler Interaktionspunkt; VAFIP = verborgener anwendungsfunktionaler Interaktionspunkt; WAFIP = wahrnehmbarer anwendungsfunktionaler Interaktionspunkt.]

Oesterreich ([27] S.121) leitet aus handlungstheoretischen Konzepten ab, daß ein Handelnder seine Zielkonsequenzen stets so bestimmt, "daß er möglichst große Freiheit hat, beliebig weitere Zielkonsequenzen anzustreben. ... Der Nutzen ist um so höher, je mehr ein Ereignis erlaubt, im folgenden Wahlfreiheit zwischen mehreren Wegen der Fortsetzung des Handelns zu haben, vorausgesetzt, diese Wege sind auch effektiv." Benutzer sollten also bevorzugt Dialogkontexte mit einer großen dialog- bzw. anwendungsbezogenen Flexibilität aufsuchen, um so ihre prospektiven Freiheiten bzgl. möglicher Handlungsalternativen zu optimieren. Effektiv ist ein Dialogkontext allerdings erst dann, wenn er auch alle aufgabenrelevanten Anwendungsfunktionen enthält.

Zur Berechnung der durchschnittlichen lokalen Flexibilität werden die beiden folgenden Maße definiert: *Anwendungsflexibilität* (AFI) für das Handlungsfeld der Anwendungsoperatoren und *Dialogflexibilität* (DFI) für das

Handlungsfeld der Dialogoperatoren. Die Anzahl an ausführbaren Operatoren im jeweiligen Dialogkontext wird als lokale Flexibilität definiert.

$$AFI = \frac{1}{K} \sum_{d=1}^K \# VAFIP_d$$

AFI relatives Maß für die anwendungsbezogene Flexibilität (lokal);

VAFIP_d Anzahl an allen verborgenen anwendungsbezogenen funktionalen Interaktionspunkte im Dialogkontext d;

K Anzahl der analysierten Dialogkontexte D.

DFI wird genauso berechnet wie AFI, nur daß statt VAFIP die Menge VDFIP zur Anwendung kommt.

Ein interaktives System sollte also dann besonders benutzungsgerecht sein, wenn es ein hohes Maß an AFI (eventuell auch an DFI) aufweist. Und genau diese Hypothese läßt sich mit empirischen Studien untersuchen. Wie werden nun die Kennwerte für eine gegebene Benutzungsoberfläche ermittelt? Die folgende *Verfahrensvorschrift* beschreibt, wie im einzelnen vorzugehen ist:

1. Es werden alle zu analysierenden Dialogkontexte festgelegt und in Form eines Interaktionsstrukturschemas graphisch aufbereitet (siehe Abbildung 7). Ein Dialogkontext ist im wesentlichen durch die Menge der aktuell ausführbaren Operatoren definiert. Jeder mögliche Wechsel zwischen einzelnen Dialogkontexten wird durch eine gerichtete Kante in dieses Schema eingetragen (siehe [11], [41], [3], [16], sowie [22]). [*Anmerkung*: die Menüoptionen von Pull-Down- bzw. Pop-Up-Menüs werden nicht als eigenständige Dialogkontexte gezählt, sondern dem globalen Dialogkontext als WFn zugeordnet. Bei einer wahrnehmbaren Funktion (WF) für die Eingabe über Tastatur werden alle relevanten ASCII- und Funktionstasten als jeweils eine verborgene Funktion (VF) gezählt; dies gilt nicht für Kommandooberflächen, weil dort nur alle möglichen Operatoren – und nicht alle Aktionen – gezählt werden.]
2. Für jeden Dialogkontext wird die Anzahl aller wahrnehmbaren und verborgenen interaktiven Funktionen ausgezählt und in das zugehörige Schema des Dialogkontextes eingetragen. Es ist jeweils festzustellen, ob es sich dabei um eine Dialog- oder Anwendungsfunktion handelt.
3. Für jeden Dialogkontext wird das Verhältnis wahrnehmbarer Funktionen (WF) zu verborgenen Funktionen (VF) berechnet und dem entsprechenden Dialogkontext zugeordnet. Dies geschieht ebenso mit der Anzahl wegführender Kanten zur Berechnung des interaktiven Verzweigungsgrades (IVG).

4. Der produktbezogene Kennwert des jeweiligen Maßes errechnet sich gemäß der entsprechenden Formel über alle der Analyse zugrunde liegenden Dialogkontexte D.

5. Empirische Validierungstudien

Insgesamt wurden drei unterschiedliche Programme mit jeweils zwei verschiedenen Oberflächen analysiert und mit den fünf Metriken quantifiziert. Für jedes dieser drei Programme gibt es mindestens eine empirische Vergleichsstudie, in der die beiden verschiedenen Oberflächen miteinander verglichen wurden. Die ausführlichen Beschreibungen der Programme und der Vergleichsstudien können in den angegebenen Quellen nachgelesen werden. Hier können nur die wesentlichen Ergebnisse aufgeführt und diskutiert werden.

5.1. Test-I: CUI versus GUI

Es wurde eine traditionelle, zeichenorientierte Oberfläche ('character oriented user interface' CUI; siehe Abbildung 8; [1]) mit einer weitgehend hierarchischen Dialogstruktur gegen eine Desktop-Oberfläche ('graphical user interface' GUI; siehe Abbildung 9; [2]) eines relationalen Datenbankprogrammes getestet ([30] und [33]).

In diesem Vergleichstest hatten sowohl sechs Anfänger als auch sechs Experten die Aufgabe, jeweils mit der ihnen zugeordneten Oberfläche zehn verschiedene, unterschiedlich komplexe Datenbankaufgaben zu bearbeiten. Da die Anfänger nur die ersten sechs Aufgaben bewältigt haben, werden hier die Ergebnisse nur dieser Aufgaben dargestellt (ausführlichere Darstellung in [33]). Insgesamt nahmen 24 Testpersonen in Einzelsitzungen teil (12 Testpersonen pro Oberfläche).

Es zeigten sich nicht nur für Anfänger, sondern auch für Experten signifikante Performanzvorteile (Signifikanzniveau: $p \leq .050$, siehe Tabelle 4). Die GUI-Oberfläche weist geringere Bearbeitungszeiten auf.

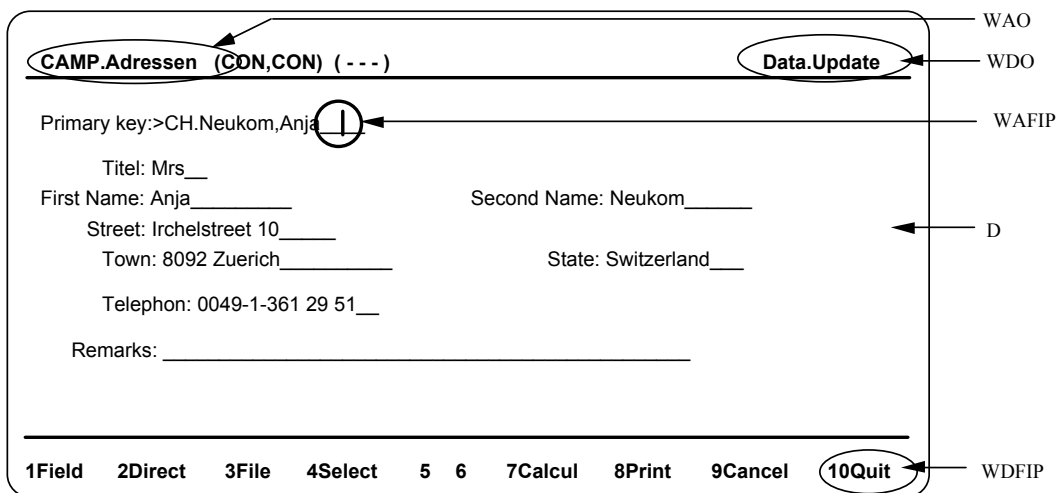


Abbildung 8: Die zeichenorientierte Benutzungsoberfläche (CUI) mit beispielhaft gekennzeichneten Elementen des Objekt- und Funktionsraumes.

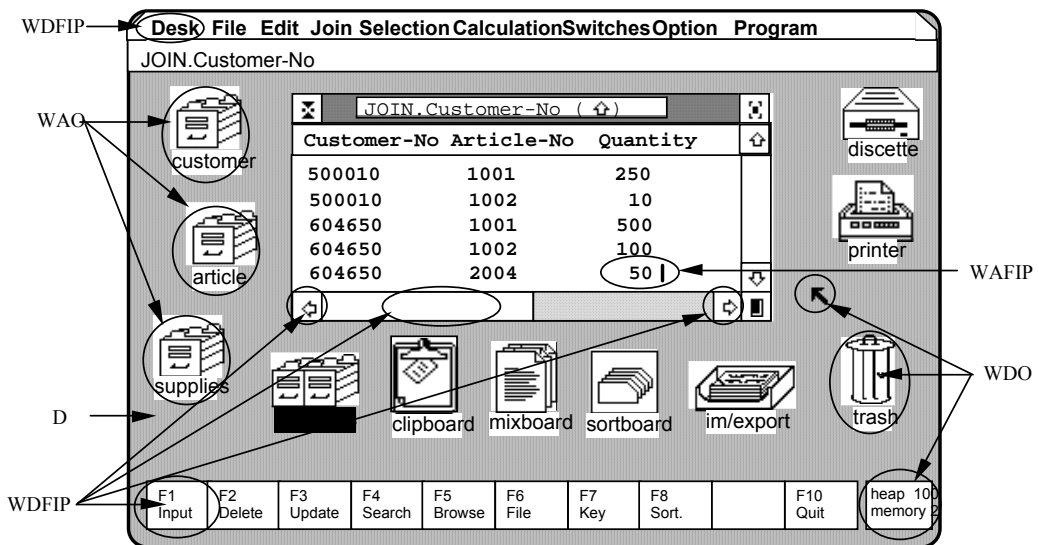


Abbildung 9: Die graphische Benutzungsoberfläche (GUI) mit beispielhaft gekennzeichneten Elementen des Objekt- und Funktionsraumes.

Tabelle 4: Ergebnis für Test-I: varianzanalytische Auswertung für die Performanz gemessen über die Aufgabenbearbeitungszeit (Angaben aus [33]).

Abhängige Meßgröße: 'Aufgabenbearbeitungszeit'	Mittelwert ± Standardabweichung		Signifikanz <i>p</i>
	CUI (N=6)	GUI (N=6)	
Anfänger	1073 s ± 590 s	670 s ± 490 s	< .002
Experten	414 s ± 245 s	201 s ± 137 s	< .001
Insgesamt	683 s ± 556 s	418 s ± 437 s	< .001

5.2. Test-II: multimediale Oberfläche mit hierarchischer Dialogstruktur versus multimediale Oberfläche mit netzartiger Dialogstruktur

Es wurden zwei multimediale Oberflächen (GUI, siehe Abbildung 10) eines Bankeninformationssystems [9] gegeneinander getestet [6].

Sechs Anfänger hatten mit jeder Oberfläche zehn verschiedene, einfache Suchaufgaben zu lösen. Die beiden Oberflächen zeichneten sich dadurch aus, daß beide Oberflächen mindestens 100% Feedback besaßen (höhere Feedbackwerte kommen dadurch zustande, daß für dieselbe verborgene Funktion mehr als eine Repräsentation in dem jeweiligen Dialogkontext vorhanden ist). Beide Oberflächen unterschieden sich ausschließlich

durch ihre Dialogstruktur. Die netzartige Dialogstruktur (GUI_{netz}) wurde im Unterschied zur hierarchischen Dialogstruktur (GUI_{hier}) so konstruiert, daß sich ein – im Vergleich zu CUI versus GUI aus Test-I – möglichst identisches *Flexibilitätsverhältnis* ergab, jedoch die konkreten Kennwerte in ihrer absoluten Ausprägung deutlich kleiner ausfielen.

Die Gebrauchstauglichkeit wurde mit zwei Performanzmaßen (abhängige Meßgrößen) gemessen: (1) Aufgabenbearbeitungszeit und (2) Anzahl Maskenwechsel. Weder hinsichtlich der Aufgabenbearbeitungszeit, noch hinsichtlich der getätigten Maskenwechsel ergeben sich empirisch nachweisbare Performanzvorteile für die flexiblere Oberfläche ($p \leq .085$ sowie $.625$; siehe Tabelle 5).



Abbildung 10: Die multimediale Benutzungsoberfläche (GUI) mit beispielhaft gekennzeichneten Elementen des Objekt- und Funktionsraumes.

Tabelle 5: Ergebnis für Test-II: varianzanalytische Auswertung für die Performanz gemessen über die Aufgabenbearbeitungszeit (Angaben aus [6]).

Abhängige Meßgröße:	Mittelwert ± Standardabweichung		Signifikanz
	GUI_{hier} (N=6)	GUI_{netz} (N=6)	
'Aufgabenbearbeitungszeit'	9.7 min ± 3.8 min	10.8 min ± 4.3 min	< .085
'Anzahl Maskenwechsel'	54 ± 15 Masken	56 ± 19 Masken	< .625

5.3. Test-III: CUI mit hierarchischer versus CUI mit netzartiger Dialogstruktur

In dem Vergleichstest von Grützmaier [18] wurden zwei zeichenorientierte Oberflächen mit unterschiedlich flexiblen Dialogstrukturen getestet (siehe Abbildung 11).

Insgesamt hatten 35 anonyme Testpersonen mit einem im Universitätsrechenzentrum öffentlich zugänglichen Simulationsprogramm versucht, den in der beigelegten Instruktionsdokumentation angegebenen Zielkriterien (Soll-Werten) möglichst nahe zu kommen. Mittels automatischer Protokollierung wurden die Ist-Werte aller

Parameter bei Simulationsende festgehalten. Die abhängige Meßgröße zur Messung von Performanzunterschieden besteht aus einem integrierten Maß über acht verschiedene Soll-Ist-Wertdifferenzen in Form von Absolutwerten, welches den prozentualen Zielerreichungsgrad mißt. Insgesamt standen den Testpersonen 49 verschiedene Parameter in über dreihundert verschiedenen Dialogkontexten (Masken) zur Steuerung der Simulation zur Verfügung. Es ergab sich kein meßbarer Performanzunterschied ($p \leq .825$; siehe Tabelle 6).

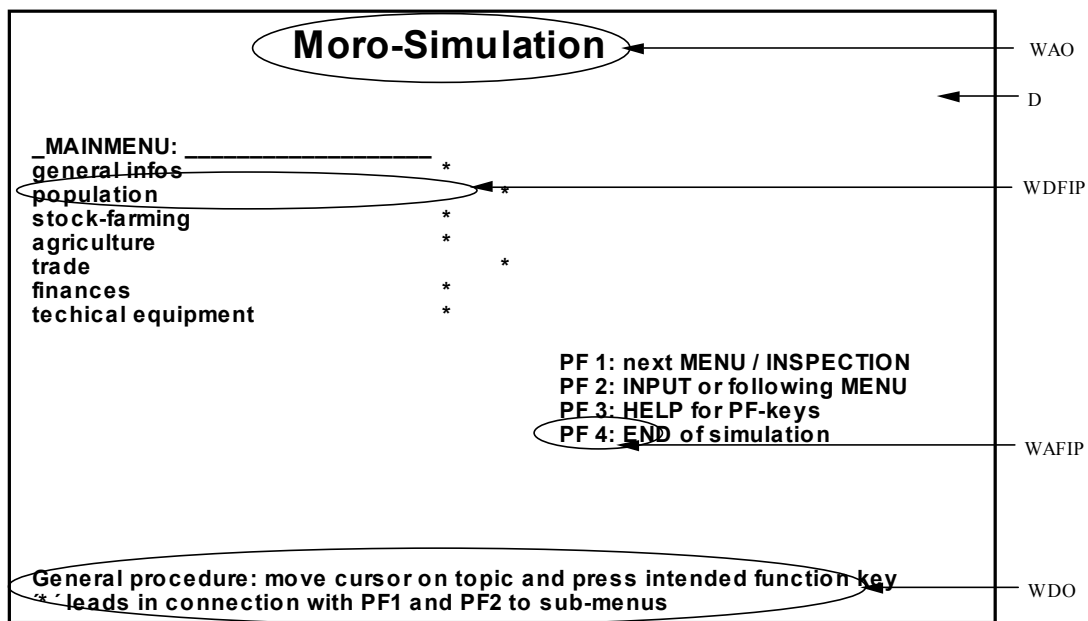


Abbildung 11: Die zeichenorientierte Benutzungsoberfläche (CUI) des Simulationsprogrammes mit beispielhaft gekennzeichneten Elementen des Objekt- und Funktionsraumes.

Tabelle 6: Ergebnisse für Test-III: varianzanalytischen Auswertung für die absolute Zielabweichung (Angaben aus [18] S. 55)

Abhängige Meßgröße:	Mittelwert ± Standardabweichung		Signifikanz
	$CUI_{hier} (N=20)$	$CUI_{net} z(N=15)$	p
'Zielabweichung'	49% ± 15%	48% ± 18%	< .825

5.4. Diskussion

Wie lassen sich diese empirischen Befunde erklären? Welche produktspezifischen Eigenschaften sind für das teilweise unterschiedliche Abschneiden der jeweiligen Oberflächen ausschlaggebend?

Es zeigte sich in den drei experimentellen Vergleichsstudien lediglich ein nachweisbarer Vorteil für die GUI-Oberfläche von Adimens (Test-I; $p \leq .050$; siehe Tabelle 4). In den beiden anderen Vergleichstests

ergeben sich keine empirisch nachweisbaren Unterschiede zwischen den unterschiedlichen Interaktionsstrukturen ($p \geq .050$; siehe Tabelle 5 für den Test-II und Tabelle 6 für den Test-III).

Wenn sich nun hinsichtlich des einen oder anderen Kennwertes – gemessen über die hier vorgestellten Metriken –, ein erkennbarer Unterschied zwischen den verschiedenen Oberflächen als durchgängig und konstant

herausstellt, dann hätte man einen ersten Anhaltspunkt für die Validierung der hier vorgestellten Maße.

6. Kennwerte der fünf Metriken für Feedback und Flexibilität

Um die Fragen nach der Interpretierbarkeit der empirischen Befunde beantworten zu können, wurden die definierten Metriken auf alle sechs Oberflächen angewandt. Es ergeben sich folgende Kennwerte für die beiden Metriken bzgl. des Ausmaßes an *Feedback* (siehe Tabelle 7).

Die CUI-Oberfläche zeichnet sich gegenüber der GUI-Oberfläche durch etwas mehr visuelles Feedback aus (CUI := 73%, GUI := 66%; siehe Tabelle 7). Die multimedialen Oberflächen sind praktisch identisch hinsichtlich der Anzahl an wahrnehmbaren Repräsentationen der implementierten Funktionalität. Die beiden CUI-Oberflächen des Simulationsprogrammes von Grützmaier (1988) sind weitgehend vergleichbar bzgl. des Ausmaßes an visuellem Feedback. Deutliche Unterschiede treten jedoch bei der Quantifizierung der

Flexibilität der verschiedenen Interaktionsstrukturen auf (siehe Tabelle 8).

Die höchsten Kennwerte für die globale Flexibilität (IVG) zeigen sich bei den beiden multimedialen Oberflächen. Dies kommt dadurch zustande, daß es praktisch keine dialogkontextbewahrenden Operatoren gab. Dies drückt sich auch in den niedrigen Werten von AFI und DFI aus (siehe Tabelle 8).

Die netzartige CUI-Oberfläche des Simulationsprogrammes hat einen mehr als doppelt so großen Kennwert für IVG (=3,4) wie die hierarchische CUI-Oberfläche (IVG=1,5). Die GUI-Oberfläche von Adimens weist gegenüber der CUI-Oberfläche deutlich höhere Kennwerte hinsichtlich DFI und AFI auf. Im Unterschied zu allen anderen Kennwerten für AFI und DFI liegen diese beiden Kennwerte der GUI-Oberfläche über 12 (AFI = 19,5 und DFI = 20,4; siehe Tabelle 8). Der mittlere Wert (15) zwischen 10 und 20 kann als mögliche Grenze gesetzt werden, oberhalb derer sich die im Sinne dieser Metrik gemessene Flexibilität auch empirisch nachweisen läßt. Weitere Forschung wird zeigen müssen, ob dieser Grenzwert nicht größer ist und näher bei 20 liegt.

Tabelle 7: Übersicht über die Kennwerte der Feedbackmaße AFBF und RFBF für die sechs verschiedenen untersuchten Oberflächen [K = Anzahl analysierter Dialogkontexte].

Oberflächentyp	AFBF	RFBF	K
CUI-Oberfläche	26,2	73%	36
GUI-Oberfläche	18,4	66%	28
Multimedia-Oberfläche: hierarchisch	68,0	100%	68
Multimedia-Oberfläche: netzartig	66,9	103%	65
CUI-Oberfläche: hierarchisch	327,8	86%	435
CUI-Oberfläche: netzartig	347,9	90%	388

Tabelle 8: Übersicht über die Kennwerte der Flexibilitätsmaße IVG, DFI und AFI für die sechs verschiedenen untersuchten Oberflächen [K = Anzahl Dialogkontexte].

Oberflächentyp	IVG	Ratio	AFI	Ratio	DFI	Ratio	K
CUI-Oberfläche	1,8		12,1		10,1		36
GUI-Oberfläche	2,4	0,75	19,5	0,62	20,4	0,50	28
Multimedia-Oberfläche: hierarchisch	4,1		3,6		0,5		68
Multimedia-Oberfläche: netzartig	5,7	0,72	4,2	0,86	1,3	0,38	65
CUI-Oberfläche: hierarchisch	1,5		1,9		1,9		435
CUI-Oberfläche: netzartig	3,4	0,44	1,3	1,46	2,7	0,70	388

7. Diskussion

Es wurde ein allgemein anwendbares Beschreibungskonzept für Benutzungsoberflächen vorgestellt, welches einerseits alle Interaktionsfunktionen in Anwendungs-

funktionen und Dialogfunktionen aufteilt, und andererseits – aus Sicht von Benutzern – zwischen wahrnehmbaren und verborgenen Interaktionsobjekten bzw. -funktionen unterscheidet. Aufbauend auf diesen abstrakten Beschreibungskonstrukten lassen sich für die

meisten gängigen Oberflächentypen einige handlungspsychologisch relevante Aspekte erkennen und auszählen. Zur Berechnung der softwareergonomischen Produktgüte werden fünf Metriken für 'Feedback' und 'Flexibilität' definiert und anhand von drei experimentellen Vergleichsstudien validiert.

Wie man aus Tabelle 7 und Tabelle 8 erkennen kann, läßt sich ein empirischer Performanzvorteil nur dann für eine Oberfläche nachweisen, wenn die Interaktionsstruktur gemessen mittels DFI und AFL größer als 15 ist (siehe GUI-Oberfläche in Tabelle 8). Dabei ist offenbar ein Unterschied bei dem interaktiven Verzweigungsgrad (IVG) als Maß für *globale* Freiheitsgrade nicht so bedeutsam wie der Unterschied bei dem *lokalen* Ausmaß an Freiheitsgraden (DFI und AFL). Das Ausmaß an visuellem Feedback ist *nicht* für den empirisch nachweisbaren Unterschied CUI vs. GUI des Datenbankprogrammes von primärer Bedeutung.

Wenn man den Schwellwert von [DFI = 15] als Kriterium auf die Interaktionsstrukturen der Vergleichsstudie von Kühn und Streitz [24] nachträglich anwendet, so kann man das Ergebnis dieser empirischen Vergleichsstudie *vorhersagen*. Nimmt man an, daß die in der Veröffentlichung abgebildeten Interaktionsstrukturen eine vollständige Darstellung sind, so ist DFI für die abgebildete 'lowfan'-Struktur DFI = 4,7 und für die 'highfan'-Struktur DFI = 7,0. Diese Kennwerte liegen deutlich unter dem in dieser Arbeit postulierten Schwellwert von 15. Die durchschnittliche Aufgabenbearbeitungszeit der 145 Aufgaben für die 'lowfan'-Oberfläche beträgt 118,4 Sekunden und für die 'highfan'-Oberfläche 115,4 Sekunden. Dieser Performanzunterschied ist – wie nachträglich 'vorhergesagt' – *nicht* signifikant.

Eine Reihe von Vergleichsstudien zwischen verschiedenen Menüstrukturen haben gezeigt [29], daß die Benutzbarkeit der Menübäume von ihrer 'Breite' (= Anzahl alternativer Menüoptionen pro Menü) und ihrer 'Tiefe' (= Anzahl Hierarchiestufen) abhängt. Man kann nun annehmen, daß die Anzahl Menüoptionen eine – wenn auch eingeschränkt gültige – Metrik für die interaktive Flexibilität der gesamten Menüstruktur ist. Paap und Roske-Hofstrand ([29] S. 222) können anhand verschiedener empirischer Vergleichsstudien aufzeigen, daß für eine optimale Benutzung die minimale Anzahl mindestens 16 Menüoptionen umfaßt, sofern diese Optionen sinnvoll gruppiert auf der Maske dargeboten werden. Diese Resultate können als eine weitere Bestätigung des Minimalwertes von 15 für DFI bzw. AFL angesehen werden.

Es ist wahrscheinlich für die beiden Flexibilitätsmaße DFI und AFL gelungen, Mindestwerte bzw. Schwellwerte quantitativ angeben zu können. Die Angabe von Mindestwerten erlaubt es, statt bisher nur von einer Richtlinie nun von einem Kriterium zu sprechen, dessen Erfüllungs- bzw. Einhaltungsgang eindeutig festgestellt werden kann. Erst, wenn ein interaktives System diesen

Mindestwert erfüllt, kann wahrscheinlich von einem empirisch nachweisbaren Vorteil aufgrund hinreichender interaktiver Flexibilität ausgegangen werden.

Oberflächen lassen sich nun hinsichtlich einiger Aspekte in ihrem Ausmaß an Feedback und Flexibilität quantitativ beschreiben und damit eindeutig klassifizieren. Festzustellen, inwieweit dieser Ansatz grundsätzlich weiterführt, bleibt zunächst weiterer Forschung in dieser Richtung vorbehalten. Das in dieser Arbeit vorgestellte Vorgehen ist ein Beitrag zur Entwicklung von objektiven Meßvorschriften für die Bestimmung der softwareergonomischen Produktgüte interaktiver EDV-Systeme.

Danksagung: An dieser Stelle möchte ich mich sehr herzlich bei meinen Kollegen des BOSS-Projektes und der ADI GmbH bedanken, ohne deren Hilfe diese Arbeit nicht möglich gewesen wäre.

8. Literaturverzeichnis

- [1] ADIMENS (1988) Adimens Manual Version 2.21. ADI Software GmbH, Hardeckstr. 5, D-76185 Karlsruhe (D).
- [2] ADIMENS (1990) Adimens GTplus Manual Version 3.0. ADI Software GmbH, Hardeckstr. 5, D-76185 Karlsruhe (D).
- [3] ALTY J. L. & MULLIN J. (1989) Dialogue specification in the GRADIENT dialogue system. In: A. Sutcliffe & L. Macaulay (eds.) People and Computers V. Cambridge: Cambridge University Press, pp. 151-168.
- [4] APPLE (1987) Human Interface Guidelines: The Apple Desktop Interface. Reading: Addison-Wesley.
- [5] BALZERT H. (1989) Die Entwicklung von Software-Systemen. (Reihe Informatik, Band 34, K.H. Böhling, U. Kulisch & H. Maurer, Hrsg.) Mannheim: Bibliographisches Institut.
- [6] BRUNNER M. & RAUTERBERG M. (1993) Hierarchische oder netzartige Interaktionsstruktur bei multimedialen Informationssystemen: eine experimentelle Vergleichsstudie. Technical Report MM-2-93. Institut für Arbeitspsychologie, Zürich: ETH.
- [7] COCKTON G. (1991) Human factors and structured software development: the importance of software structure. In: D. Diaper & N. Hammond (eds.) People and Computers VI: Proceedings of the Sixth conference of the British Computer Society. Cambridge: Cambridge University Press, pp. 57-72.
- [8] COUTAZ J. (1989) UIMS: promises, failures and trends. In: A. Sutcliffe & L. Macaulay (eds.) People and Computers V: Proceedings of the Fifth conference of the British Computer Society. Cambridge: Cambridge University Press, pp. 71-84.
- [9] DAUM W. & SCHLAGENHAUF K. (1993) Mehr verkaufen, top präsentieren und informieren. Betriebswirtschaftliche Blätter 3: III-VIII.
- [10] de HAAN G., van der VEER G. & VLIET J. (1991) Formal modelling techniques in human-computer interaction. Acta Psychologica 78:27-67.
- [11] DENERT E. (1977) Specification and Design of Dialogue Systems with State Diagrams. In: E. Morlet & D.

- Ribbens (eds.) International Computing Symposium '77. Amsterdam: Elsevier, pp. 417-424.
- [12] DIN 66 234 Teil 8 (1988) Bildschirmarbeitsplätze - Grundsätze ergonomischer Dialoggestaltung. Deutsche Industrie Norm, Beuth Verlag, BRD-1000 Berlin 30.
- [13] DIX A., FINLAY J., ABOWD G. & BEALE R. (1995) Mensch Maschine Methodik. New York München: Prentice Hall.
- [14] EDMONDS E. & HAGIWARA N. (1990) An experiment in interactive architectures. In: D. Diaper, D. Gilmore, G. Cockton & B. Shackel (eds.) Human-Computer Interaction - INTERACT '90. Amsterdam: Elsevier Science, pp. 601-606.
- [15] ENGLISCH J. (1993) Ergonomie von Softwareprodukten. (Angewandte Informatik, Band 5; H. Balzert, Hrsg.), Mannheim: Wissenschaftsverlag.
- [16] GIESKENS D. F. & FOLEY J. D. (1992) Controlling user interface objects through pre- and postconditions. In: P. Bauersfield, J. Bennett & G. Lynch (eds.) Human Factors in Computing Systems CHI'92. New York: ACM, pp. 189-194.
- [17] GILMORE D. (1991) Visibility: a dimensional analysis. In: D. Diaper & N. Hammond (eds.) People and Computers VI. (British Computer Society Conference Series 3), Cambridge: Cambridge University Press, pp. 317-329.
- [18] GRÜTZMACHER A. (1988) Datenpräsentation und Lösungsverhalten in einer komplexen, simulierten Problemsituation. unveröffentlichte Lizentiatsarbeit. (Philosophische Fakultät I, Psychologisches Institut, Abteilung Angewandte Psychologie), Zürich: Universität Zürich.
- [19] HACKER W. (1986) Arbeitspsychologie. Bern: Huber.
- [20] HÜBNER W. (1990) Entwurf graphischer Benutzerschnittstellen. Berlin: Springer.
- [21] IBM (1992) Object-Oriented Interface Design. Carmel: Que.
- [22] JANSSEN C. (1993) Dialognetze zur Beschreibung von Dialogabläufen in grafisch-interaktiven Systemen. In: K. Rödiger (Hrsg.) Software-Ergonomie '93. (German Chapter of the ACM Berichte, Band 39), Stuttgart: Teubner, S. 67-76.
- [23] KISHI N. & KINOE Y. (1991) Assessing usability evaluation methods in a software development process. In: Bullinger, H-J (ed.) Human Aspects in Computing: Design and Use of Interactive Systems and Work with Terminals. Amsterdam : Elsevier. 597-601.
- [24] KÜHN E. & STREITZ N. (1989) Do users know how to ACT* when menus are complex? - A numerical model of user interface complexity. In: F. Klix, N. Streitz, Y. Waern and H. Wandke (eds.) Man-Computer Interaction Research - MACINTER II. Amsterdam: North-Holland; pp. 361-374.
- [25] MICROSOFT (1991) The Windows Interface. Redmond: Microsoft Press.
- [26] MÜLLER-HOLZ B., HACKER S. & BARTSCH T. (1990) PROTOS – Entwicklung von Methoden zur Herstellung und Bewertung von Prototypen für Benutzerschnittstellen. Zwischenbericht 8/90. Lehrstuhl für Psychologie der TU München.
- [27] OESTERREICH R. (1982) Der Begriff 'Effizienz-Divergenz' als theoretischer Zugang zu Problemen der Planung des Handelns und seiner Motivation. In: W. Hacker, W. Volpert & M. von Cranach (Hrsg.) Kognitive und motivationale Aspekte der Handlung. Bern: Huber, S. 110-122.
- [28] OPPERMANN R., MURCHNER B., PAETAU M., PIEPER M., SIMM H. & STELLMACHER I. (1988) Evaluation von Dialogsystemen - der softwareergonomische Leitfaden EVADIS. Berlin: de Gruyter.
- [29] PAAP K. R. & ROSKE-HOFSTRAND R. J. (1988) Design of menus. In: M. Helander (ed.) Handbook of Human-Computer Interaction. Amsterdam: North Holland, pp. 205-235.
- [30] RAUTERBERG M. (1989) Maus versus Funktionstaste: ein empirischer Vergleich einer desktop- mit einer ascii-orientierten Benutzungsoberfläche. In: S. Maaß & H. Oberquelle (Hrsg.) Software-Ergonomie '89. (German Chapter of the ACM Berichte, Band 32), Teubner: Stuttgart; S 313-323.
- [31] RAUTERBERG M. (1991) Interaktive Aufsetzpunkte: ein Konzept zur Beschreibung und Klassifizierung von Benutzungsoberflächen interaktiver Software. In: M. Frese, C. Kasten, C. Skarpelis & B. Zang-Scheucher (Hrsg.) Ergänzung zum Tagungsband "Software für die Arbeit von morgen". Bonn: BMFT-DLR/ Projektträger-schaft "AuT", S. 21-34.
- [32] RAUTERBERG M. (1992a) Lässt sich die Gebrauchstauglichkeit interaktiver Software messen? Ergonomie & Informatik 16:3-18.
- [33] RAUTERBERG M. (1992b) An empirical comparison of menu-selection (CUI) and desktop (GUI) computer programs carried out by beginners and experts. Behaviour & Information Technology 11(4): 227-236.
- [34] RAUTERBERG M., SPINAS P., STROHM O., ULICH E. & WAEBER D. (1994) Benutzerorientierte Software-Entwicklung. (Mensch-Technik-Organisation, Band 3, E. Ulich, Hrsg.) Stuttgart: Teubner.
- [35] RENGGER R., MACLEOD M., BOWDEN R., DRYNAN A. & BLAYNEY M. (1992) MUSIC Performance Measurement Handbook. National Physical Laboratory, DITC, Teddington, UK.
- [36] STREITZ N. (1990) Psychologische Aspekte der Mensch-Computer-Interaktion. In: C.G. Hoyos & B. Zimolong (Hrsg.) Ingenieurspsychologie. Göttingen Toronto Zürich: Hogrefe, S 240-284.
- [37] SWEENEY M., MACGUIRE M. & SHACKEL B. (1993) Evaluating user-computer interaction: a framework. International Journal of Man-Machine Studies 38: 689-711.
- [38] ULICH E. (1994, 3.Auflage) Arbeitspsychologie. Stuttgart: Poeschel.
- [39] WANDMACHER J. (1993) Software-Ergonomie. (Mensch-Computer-Kommunikation: Grundwissen, Band 2; H. Balzert, Hrsg.), Berlin: de Gruyter.
- [40] WHITEFIELD A., WILSON F. & DOWELL J. (1991) A framework for human factors evaluation. Behaviour and Information Technology 10(1): 65-79.
- [41] ZEMANEK G. V. (1987) Modellierung von Dialogstrukturen mittels Graphen. Technischer Report IBM; Heidelberg: IBM Wissenschaftliches Zentrum.