

# Von der Kommandooberfläche zur direkten Manipulation

Matthias Rauterberg

Institut für Arbeitspsychologie (IfAP)  
Eidgenössische Technische Hochschule (ETH)  
Nelkenstrasse 11, CH-8092 Zürich  
Email: rauterberg@ifap.bepi.ethz.ch  
WWW: <http://www.ifap.bepi.ethz.ch/~rauterberg/>

## Zusammenfassung

Aufbauend auf dem Konzept zur Quantifizierung von softwareergonomischen Richtlinien in Rauterberg (1995) werden die beiden Metriken *Visualisierungsgrad* und *interaktive Direktheit* definiert. Mittels dieser beiden Metriken lassen sich die häufigsten Arten von Benutzungsoberflächen eindeutig klassifizieren. Um diese Klassifikation zu validieren, wird eine Metaanalyse verschiedener empirischer Vergleichsstudien durchgeführt. Es zeigt sich, dass die eindeutige Überlegenheit von direktmanipulativen Oberflächen gegenüber Menü- und Kommandooberflächen sowohl durch einen hohen Visualisierungsgrad, als auch durch eine grosse interaktive Direktheit bedingt ist. Dieses Ergebnis kann mittels einer statistischen Metaanalyse abgesichert werden. Abschliessend werden die gewonnenen Erkenntnisse im Kontext der bereits bekannten Ergebnisse softwareergonomischer Forschung diskutiert und Wege für zukünftige Forschungsgebiete aufgezeigt.

**Stichworte:** Benutzungsoberfläche, direkte Manipulation, Klassifikation, Metaanalyse.

## 1. Benutzungsoberflächen

Nachdem wir ein Beschreibungskonzept für Benutzungsoberflächen vorgestellt und diskutiert haben (siehe Rauterberg 1995), möchte ich die verschiedenen Kategorien von Benutzungsoberflächen in einem historischen Rekurs beschreiben. Danach werden wir dann unser Beschreibungskonzept auf die zur Zeit gängigsten Interaktionsarten anwenden, um diese zunächst qualitativ einordnen zu können. Das im folgenden gewählte Vorgehen kann dazu dienen, die Frage "welche Benutzungsoberfläche ist die beste?" zu beantworten. Ist eine Oberfläche mit direkter Manipulation im Prinzip wirklich so gut, wie oftmals behauptet wird?

Shneiderman (1987, S. 57) unterscheidet zwischen folgenden fünf Interaktionsarten: (1.) Kommando-eingabe, (2.) Menüauswahl, (3.) Formulardialog, (4.) direkte Manipulation und (5.) natürlichsprachliche Ein/Ausgabe. Gehen wir noch weiter zurück in die Vergangenheit, so sind noch der Lochstreifen und die Stapelverarbeitung mit Lochkarten zu erwähnen (siehe Nielsen 1993, S. 50). Interaktive Arbeit am Rechner – im eigentlichen Sinne von kurzzyklischen Ein-Ausgabesequenzen

– wurde erst mit der Kommandoeingabe über Tastatur und Bildschirmgerät ab Mitte dieses Jahrhunderts möglich.

*Kommandooberflächen* (KOn) zeichnen sich dadurch aus, dass der Benutzer über die Tastatur Interaktionsoperatoren als Zeichenketten eingibt. In diesem allgemeinen Sinne haben alle interaktiven Systeme mit einer Tastatur eine KO. Im engeren Sinne werden Oberflächen dann zu den KOn gezählt, wenn die eingegebenen Zeichenketten mit einer bestimmten Taste (z.B. CR, ENTER usw.) abgeschlossen werden. KO werden aufgrund ihrer rein sequentiellen Ein- und Ausgabestruktur auch als zeilen- und zeichenorientierte Oberflächen bezeichnet. Der Benutzer muss die Syntax und Semantik der Kommando-'Sprache' beherrschen und bei der Eingabe fehlerfrei reproduzieren können, um ein interaktives System mit einer KO adäquat benutzen zu können (Oberquelle 1994, S. 119).

Als eine vereinfachte Variante von Kommandooberflächen wurden Frage-Antwortdialoge realisiert. Meistens bieten Frage-Antwortdialoge dem Benutzer die aktuellen Antwortmöglichkeiten zum Auswählen an. In diesem Fall gehören sie zu den Oberflächen mit Menüauswahl. Wenn auf einer Bildschirmmaske nicht nur eine, sondern parallel eine Reihe von Eingabestellen bzw. Interaktionspunkten vorgesehen sind, an denen der Benutzer verschiedene Operatoren eingeben kann, handelt es sich in der Regel um *Formulardialoge* (Shneiderman 1987). Oftmals wird dem Benutzer an diesen Eingabestellen eine Menüauswahl der verschiedenen Eingaben angeboten. Sind alle Eingaben im aktuellen Dialogkontext getätigt, so kann der Benutzer alle auf einmal mittels einer speziellen Operation (z.B. die Taste 'DatFrg') als 'komplexes Kommando' an den Rechner 'abschicken'. Wir betrachten daher Formulardialoge als eine Mischung aus Kommando- und Menüoberfläche.

*Menüoberflächen* (MOn) sind dadurch erkennbar, dass der Benutzer in einer Menüstruktur die einzugebenden Operatoren auswählen kann. Verschiedene MOn unterscheiden sich darin, wie diese Auswahl im Einzelnen erfolgt (ganze Menümasken, Pop-Up-, Pull-down-, Drop-down-Menüs, siehe Lauter 1987, S. 9-12). Für eine Menüauswahl ist zwingend notwendig, dass dem Benutzer eine wahrnehmbare Repräsentation der einzelnen Operatoren auf der Benutzungsoberfläche angeboten wird. Wegen der Vielzahl an Operatoren sind

hierbei Strukturierungshilfen notwendig. So gibt es baum- und netzartige Menüstrukturen mit verschiedenen Menüoptionen auf der aktuellen Menüebene. Die z.B. am häufigsten benötigten Operatoren werden oftmals direkt auf einzelne Funktionstasten der Tastatur abgebildet und stehen dem Benutzer dadurch zusätzlich parallel zur Verfügung.

Wenn die Darstellung aller Ausgaben einer MO (1.) auf einem grafikfähigen Bildschirm erfolgt, (2.) einzelne Objekte als verschiebbare Piktogramme dargestellt sind (Staufer 1987), und (3.) die Menüstruktur in Form einer Pull-down-Menüleiste gegeben ist, so spricht man auch von *Desktopoberflächen* (König 1989). Der wohl wesentlichste Unterschied zu den traditionellen MOn besteht im Wechsel von der Funktions-Objekt-Struktur (FO-Struktur) zur Objekt-Funktions-Struktur (OF-Struktur, siehe König 1989, S. 21f). Die FO-Struktur einer Oberfläche bedeutet, dass zuerst die gewünschte Funktion über den entsprechenden Operator und erst dann das zugehörige Objekt eingegeben werden. Bei der OF-Struktur ist es genau umgekehrt. In der Studie von Bannert (1991, S. 52) gaben 60 von insgesamt 64 Testpersonen an, eine 'objektorientierte' Vorgehensweise gemäss der OF-Struktur zu bevorzugen. Sind auch Pop-up-Menüs und/oder Piktogramme mit Operatorfunktionalität vorhanden, so erweitern sich Desktopoberflächen zu der umfassenderen Klasse der direktmanipulierbaren Oberflächen.

*Direktmanipulierbare Oberflächen* (DO) sind nach Shneiderman (1983) alle Oberflächen, "bei denen eine permanente Sichtbarkeit aller relevanten Objekte, Ersetzung komplexer Kommandos durch physische Aktionen, wie Mausbewegungen, Selektionsaktionen und Funktionstastenbetätigung, schnelle, umkehrbare, einstufige Benutzeraktionen mit unmittelbarer Rückmeldung" (Ziegler 1993, S. 146) gegeben ist. Eine wesentliche Eigenschaft von DON besteht darin, dass die Ausgaben auf dem Bildschirm in der Regel auch wieder direkt als Eingaben verwendet werden können ("inter-referential input-output", Smith et al. 1982).

Wenn bei einer DO *keine* standardisierten Vorgaben für die Ausgabe gegeben sind (im Gegensatz z.B. zu SAA/CUA, OSF/Motif, Windows usw.) und akustische und/oder haptische Sinneskanäle des Benutzers in die Interaktion einbezogen werden, so spricht man im allgemeinen von multimedialen Oberflächen.

*Multimediale Oberflächen* (MMO) bieten dem Benutzer zusätzlich zur visuellen Darstellung (Text, Graphik, Bild) auch akustische Ausgaben (Musik, Geräusche, Sprache; siehe Dannenberg und Blattner 1992). Weitere Interaktionskanäle können auch haptischer bzw. taktiler Natur sein. Allgemein ist der Gestaltungs- und Interaktionsbereich wesentlich offener als bei den traditionellen graphischen Oberflächen. Bei der Gestaltung einer MMO steht die realitätsgerechte, anschauliche Umsetzung der Anwendungsobjekte im Vordergrund (Koller 1992). Zur besonderen Veranschaulichung dynamischer Vorgänge können z.B. auch Film- bzw. Videosequenzen eingesetzt werden.

Wenn der Benutzer auf das multimediale System neben den direktmanipulativen Interaktionsarten auch über die Sprache einwirken kann, dann spricht man eher von natürlichsprachlichen Oberflächen.

*Natürlichsprachliche Oberflächen* (NO) zeichnen sich in der Regel allein durch die Tatsache aus, dass der Benutzer mit dem System über das natürliche geschriebene oder gesprochene Wort interagieren kann. Wenn nur das geschriebene Wort zugelassen ist, dann ist dies Oberfläche einer Kommandooberfläche recht ähnlich, wobei die Syntax und Semantik der Interaktion an die natürliche Sprache angelehnt ist (Vossen et al. 1991). Bei dem gesprochenen Wort ist die automatische Spracherkennung zur Zeit noch recht begrenzt. Die zur Zeit besten Spracherkennungssysteme haben bei unbeschränkter Eingabe noch eine Fehlerrate von ca. 50% (Zue et al. 1991).

Als *objektorientierte Oberflächen* (OO) sollen alle Arten von Benutzungsoberflächen zusammengefasst werden, bei denen der Benutzer direkt, möglichst mittels analoger Operationen auf hoch animierten, 2<sup>1/2</sup> oder 3D Darstellungen der Anwendungsobjekte nach der OF-Struktur interagiert. Hierunter fallen dann auch Oberflächen aus dem Bereich 'virtual reality', bei der der Benutzer sich in der Welt der Objekte selbst bewegt und mit diesen Objekten in einem virtuellen Raum interagiert. Es gehören zusätzlich zu der Klasse der OO auch die Oberflächen aus dem Bereich 'embedded virtuality' (Wellner 1993). Für ausgewählte Funktionen können manchmal natürlichsprachliche Eingaben zur Verfügung stehen (siehe als Beispiel Fujita et al. 1993). Diese Definition ist umfassender als bei Zeidler und Zellner (1992, S. 108ff).

Die verschiedenen Kategorien von Benutzungsoberflächen lassen sich nach unterschiedlichen Aspekten klassifizieren:

- (1) Zeilen- vs. Vollschirmeingabe,
- (2) system- vs. benutzerkontrolliert,
- (3) zeichen- vs. graphikorientiert,
- (4) diskrete vs. analoge Eingabe- und/oder Ausgabe,
- (5) Erinnern und Reproduzieren vs. Wiedererkennen und Auswählen,
- (6) FO-Struktur vs. OF-Struktur (siehe Kunkel, Bannert und Fach 1995),
- (7) freier vs. modaler Dialog (Zeidler und Zellner 1992, S. 96ff), sowie
- (8) deiktische vs. symbolische Referenzierung (Ziegler 1987 und Ziegler 1993).

Da sich oftmals keine der oben beschriebenen sechs Oberflächenklassen eindeutig nur einem dieser acht Klassifikationskriterien zuordnen lässt, wurden Oberklassen eingeführt: Zeichenorientierte (CUI: 'character oriented user interface') vs graphische Oberflächen (GUI: 'graphical user interface' nach dem WYSIWYG-Prinzip 'What you see is what you get'), direktmanipulierbare Oberflächen (DM bzw. DO), hybride Dialogformen (WIMP: 'windows', 'icons', 'mouse' mit OF-Struktur), objektorientierte Oberflächen.

## 1.1 Interaktionspunkte von Kommandooberflächen

Als erstes Beispiel soll die Kommandooberfläche von MsDOS™ dazu dienen, das Beschreibungskonzept der Interaktionspunkte zu verdeutlichen. Bei einer Kommandooberfläche werden die verborgenen Funktionen über die Eingabe von Interaktionsoperatoren ('Kommandos') angesprochen, welche sich aus Zeichenketten zusammensetzen.

Der Aktionsraum des Benutzers ist die Tastatur. Der grösste Teil des Bildschirms dient lediglich als Ausgabemedium. Zunächst wählen wir verschiedene dialog- und anwendungsbezogene Funktionen aus: Die Kommandos 'dirCR' und 'cd <path>CR' sind stellvertretend für alle verborgenen dialogfunktionalen Interaktionspunkte (VDFIPe), und die beiden Kommandos 'delete <file name>CR' und 'rename <old name> <new name>CR' sind stellvertretend für alle anwendungsfunktionalen Interaktionspunkte (VAFIPe).

Dem Benutzer steht in der Regel bei einer Kommandooberfläche nur *ein* wahrnehmbarer funktionaler Interaktionspunkt (WFIP) zur Verfügung: der Eingabebereich hinter dem 'Systemprompt' auf dem Bildschirm. Es besteht somit zwischen dem WFIP auf dem Bildschirm und den verborgenen VFIPen eine (1 zu n)-Beziehung. Wenn bestimmte Funktionen zusätzlich über spezielle Funktionstasten oder Tastenkombinationen ausgelöst werden können, so sind ihre Repräsentationen auf der Tastatur ebenfalls wahrnehmbare Funktionsrepräsentationen (WFIPe). Diese wahrnehmbaren Funktionsrepräsentationen (WFIP) sind z.B. über spezielle Funktionstasten (z.B. 'PRT', 'DEL') gegeben.

An einem einfachen Beispiel lässt sich die Menge der verborgenen dialogfunktionalen Interaktionspunkte (VDFIPe) verdeutlichen: Ein Teil der Anwendungsobjekte von MsDOS™ sind die auf der Festplatte bzw. Diskette verwalteten Dateien; diese Anwendungsobjekte lassen sich über Eigenschaften wie 'Name.Extension', 'Grösse in Byte', 'Datum der Erstellung', 'Datum der letzten Änderung', 'Ort der Speicherung' usw. beschreiben bzw. auf der Ein/Ausgabeschnittstelle (EAS) repräsentieren und gegebenenfalls verändern. Alle diese Eigenschaften sind jedoch nur potentiell wahrnehmbar. Erst durch den Dialogoperator 'dirCR' mit dem entsprechenden dialogfunktionalen Interaktionspunkt (VDFIP) lassen sich diese Eigenschaften in eine passiv wahrnehmbare Repräsentationsform überführen. Wie ausserordentlich hilfreich dies ist, belegen empirische Untersuchungen (Greenberg und Witten 1988; Rauterberg 1991b). Das mit Abstand am häufigsten benutzte Kommando bei der Benutzung von UNIX™ ist das 'ls' Kommando, welches wie das 'dir' Kommando bei MsDOS™ lediglich den aktuellen 'Directory'-Inhalt auf den Bildschirm ausgibt und somit eindeutig der Orientierung dient (Kraut, Hanson und Farber 1983). Es zeigte sich, dass eine der Hauptschwächen von Kommandooberflächen in der mangelnden automatischen Rückmeldung zu sehen ist. Diese Schwäche erweist sich oft-

mals nicht nur für Anfänger, sondern auch für erfahrene Benutzer als nachteilig (Kraut, Hanson und Farber 1983; Greenberg und Witten 1988).

Der Oberflächentyp 'Formularialog' ('form fill in') ist im wesentlichen eine spezielle Art einer Kommandooberfläche mit multiplen Eingabepunkten (WFIPen) im aktuellen Dialogkontext, wobei der weitere Interaktionsverlauf von den jeweils eingegebenen Werten abhängt. Das hier vorgestellte Beschreibungskonzept und die ableitbaren Gestaltungshinweise sind gleichermassen auch auf den Oberflächentyp 'Formularialog' anwendbar.

## 1.2 Interaktionspunkte von zeichenorientierten Menüoberflächen (CUI)

Sehen wir uns als zweites Beispiel die historisch nächste Entwicklungsstufe von Benutzungsoberflächen an. Eine zeichenorientierte Oberfläche ist im wesentlichen dadurch gekennzeichnet, dass möglichst für jede verborgene Funktion (VFIP) auch eine wahrnehmbare Funktionsrepräsentation (WFIP) vorhanden ist und somit dem Benutzer *Feedback* über die im aktuellen Dialogkontext gültigen Funktionen gibt (Ulich et al. 1991).

Ein besonderes Problem – insbesondere im Vergleich zu direktmanipulierbaren Oberflächen – taucht auf: Welchen repräsentationalen Stellenwert haben die Hinweise auf die aktuelle Semantik der Menüitems? Zur Beantwortung dieser Frage ist die oben eingeführte Unterscheidung in den aktiven Wahrnehmungsraum und den Aktionsraum hilfreich. Wir nennen alle semantischen Bezeichner der Menüleiste auf dem Bildschirm 'wahrnehmbare Interaktionspunkte des aktiven Wahrnehmungsraumes' und die wahrnehmbaren Interaktionspunkte der zugehörigen (Funktions)-Tasten auf der Tastatur 'repräsentationale Interaktionspunkte des Aktionsraumes'.

Da jedoch die Ein/Ausgabeschnittstelle meistens einen physikalisch begrenzten Darstellungsraum hat, können auf der Bildschirmoberfläche nur ausgewählte funktionale Interaktionspunkte repräsentiert werden (Ilg und Ziegler 1987, Sommerville 1987). Dies hat zur Konsequenz, dass die repräsentationalen Interaktionspunkte (WFIPen) auf verschiedene Dialogkontexte aufgeteilt werden müssen.

Nehmen wir an, das System MS-WORD für DOS befände sich z.B. gerade in dem Interaktionsmodus 'Befehlseingabe'. Der Benutzer hat die Möglichkeit, bei dem 'Eingabe'-WFIP (z.B. das Menüitem 'Ausschnitt') weitere Untermenüs aufzuklappen. Die Verlagerung dieses repräsentationalen Interaktionspunktes erfolgt über die Cursortasten bzw. die TAB-Taste. Alle Zeicheneingaben des Benutzers werden in diesem Zustand ausschliesslich zur Dialogsteuerung verwendet. Das aktuell ausgewählte Menüitem (z.B. 'Ausschnitt') ist in der zweit untersten Bildschirmzeile jeweils ausführlicher erläutert.

Die Unterscheidung zwischen dem aktiven Wahrnehmungsraum und dem Aktionsraum ist immer dann von

besonderer Bedeutung, wenn der für einen funktionalen Interaktionspunkt die Semantik tragende Interaktionspunkt räumlich nicht mit dem aktionalen Interaktionspunkt zusammenfällt. Dies ist dann der Fall, wenn auf dem Bildschirm die Bedeutung (in irgendeiner Form) für einzelne Tasten (insbesondere Menüeinträge oder Funktionstasten) oder sonstige Interaktionselemente als repräsentationaler Interaktionspunkt gegeben ist. Wir müssen also zwischen der Menge der *repräsentationalen Interaktionspunkte des aktiven Wahrnehmungsraumes* (WFIP) und der Menge der *repräsentationalen Interaktionspunkte des Aktionsraumes* (WFIPA) unterscheiden.

Die konzeptionelle Trennung in den aktiven Wahrnehmungsraum und den Aktionsraum ist sehr wichtig (siehe die entsprechende Unterscheidung von 'execution' – 'perception' bei Norman 1986), führte bisher jedoch leider auch zu der physikalischen Trennung dieser beiden Bereiche ('Bildschirm' – 'Tastatur'). Die physikalische Distanz zwischen diesen beiden Bereichen ist z.B. durch die räumliche Entfernung  $\Delta$  zwischen WFIP und WFIPA messbar. Um dieses Problem in der Praxis teilweise zu entschärfen, werden – bei überwiegend durch Funktionstasten gesteuerten Systemen – z.B. *Tastaturschablonen* mit ausgeliefert bzw. von den Benutzern oftmals selbst angefertigt.

Die Zuordnung zwischen dem repräsentationalen Interaktionspunkt des aktiven Wahrnehmungsraumes (WFIP) und dem repräsentationalen Interaktionspunkt des Aktionsbereiches (WFIPA) muss vom Benutzer entweder im (internen) Kurzzeit- oder bei längerer Übung im Umgang mit der jeweiligen Software auch im Langzeitgedächtnis für die handlungsleitende Entscheidung verfügbar sein. Eine andere Möglichkeit besteht darin, dass der Benutzer diese Zuordnung im externen visuellen 'Gedächtnis' (die Ein/ Ausgabeschnittstelle) belässt und die jeweilige Funktionstaste ohne Blickzuwendung hin zur Tastatur findet und betätigt. Je grösser die physikalische Distanz zwischen WFIP und WFIPA ist, desto stärker ist zumindest für den ungeübten Benutzer – aufgrund einer notwendigen Blickzuwendung zur Tastatur – die kurzzeitige Gedächtnisbelastung.

Bei direktmanipulierbaren Oberflächen wird gegenüber den CUI-Oberflächen für die maussensitiven Bereiche (bzw. die entsprechenden WFIPen) die Distanz  $\Delta$  Null sein. Dies kommt dadurch zustande, dass bei einem direktmanipulativen Interaktionselement (z.B. die 'Maus') der entsprechende WFIPA nicht die Maustaste selbst, sondern der Maus-Cursor auf dem Bildschirm ist. Dies gilt dann auch ebenso für die Cursortasten der herkömmlichen Tastatur bzw. für alle Arten der Interaktionssteuerung, bei der ein WFIPA des jeweiligen Aktionsraumes auf dem Bildschirm vorhanden ist. Hierbei wird angenommen, dass die jeweilige Positionierung bzw. Steuerung des WFIPAs auf dem Bildschirm auf der sensumotorischen Regulationsebene ausgeführt werden kann, ohne dabei mit höheren mentalen Regulationsebenen zu interagieren. Der Nachteil der Cursorsteuertasten gegenüber der Maussteuerung liegt darin begründet, dass sie – bedingt durch den

'digitalen' Charakter ihrer Steuerung – einen erhöhten zusätzlichen interaktiven Aufwand benötigen (Ilg und Ziegler 1987; Hächler et al. 1990).

### 1.3 Interaktionspunkte von direktmanipulierbaren Oberflächen (GUI)

Wenn wir sowohl einen hohen Grad an Visualisierung anstreben und gleichzeitig die interaktionale Distanz zu den funktionalen Interaktionspunkten (FIPen) minimieren wollen, kommen wir zu der Klasse der graphischen Oberflächen. Die verschiedenen Anwendungsobjekte können z.B. als Ikonen graphisch auf dem Bildschirm repräsentiert sein; die zugehörigen anwendungsbezogenen Funktionen lassen sich direkt über entsprechend zugeordnete Menüs aktivieren. Als besonders 'direkt', und damit von Benutzern bevorzugt, haben sich 'Pop-up'-Menüs in unmittelbarem raumzeitlichen Zusammenhang mit den zugehörigen Objekten erwiesen (Koller und Ziegler 1989). Wie z.B. bei der SmallTalk-Umgebung sollten für die Interaktion mehr als nur eine Maustaste zur Verfügung stehen: Eine Taste um die Objekte zu aktivieren, bewegen usw. und eine zweite Maustaste für die anwendungsbezogenen Funktionen auf diesen Objekten. Eine dritte Maustaste z.B. ermöglicht die Aktivierung der Eigenschaftsliste des aktivierten Objektes.

Eine historische Übergangsform zwischen den traditionellen CUI-Oberflächen und den direktmanipulierbaren Oberflächen bilden die Desktopoberflächen. Eine Desktopoberfläche zeichnet sich neben der grafikfähigen Bildschirmausgabe durch 'Pull-down'-Menüs und durch Ikonen, Piktogramme, Fenster, Dialogboxen, Dialogknöpfe usw. aus. Da bei grossen Bildschirmflächen die Distanz zwischen den Pull-down-Menüitems am oberen Bildschirmrand und den aktuell bearbeiteten Dialog- bzw. Anwendungsobjekten irgendwo auf dem Schirm recht gross werden kann, wurden verschiedene Interaktionstechniken (z.B. die 'Abreiss'-Menüs) entwickelt. Durch diese weiteren Dialogobjekte ergeben sich auch neue Interaktionsoperatoren.

Bei einer direktmanipulierbaren Oberfläche werden die meisten funktionalen Interaktionspunkte (FIPe) dem Benutzer direkt im aktuellen Dialogkontext in den verschiedensten Repräsentationsformen angeboten. Da bei einem einigermaßen komplexen System dennoch nicht alle funktionalen Interaktionspunkte (VFIPe) gleichzeitig als repräsentationale Interaktionspunkte (WFIPe) auf dem Bildschirm Platz haben (Ilg und Ziegler 1987, Sommerville 1987), werden sie als Menüitems z.B. bei Desktopoberflächen in den Pull-down Menüs der Menüleiste versteckt (Fabian und Rathke 1983) bzw. bei direktmanipulierbaren Oberflächen in den Pop-up-Menüs. Die Bedeutung der Tastatur für die konkrete Interaktionssteuerung nimmt bei einer direktmanipulierbaren Oberfläche stark ab und beschränkt sich meistens auf die reine Texteingabe.

Die Interaktionssteuerung bei direktmanipulierbaren Oberflächen erfolgt fast ausschliesslich über ein 'analoges' Interaktionselement (z.B. Maus, Rollkugel,

Finger bei berührungssensitiven Bildschirmen usw.; Balzert 1988), bis hin zu dem Datenhandschuh (Foley 1988). Der Vorteil dieser neuen Interaktionstechniken besteht darin, dass Aktionsraum und aktiver Wahrnehmungsraum raum-zeitlich zusammenfallen (Dumais und Jones 1985). Ebenso weisen neuere Entwicklungen im Schnittstellenbereich in diese Richtung (Ward und Phillips 1987; Mel et al. 1988). Eine Reihe von speziellen Interaktionselementen werden ausschliesslich dazu entwickelt, dem Benutzer möglichst direkt die benötigte Anwendungsfunktion darzubieten. Dies kann man z.B. bei CAD-Arbeitsplätzen beobachten, wo oftmals zusätzlich spezielle Interaktionselemente wie Drehknöpfe, Grafiktablets usw. den Benutzern zur Verfügung gestellt werden.

## 2 Oberflächenklassifikation

Das in Rauterberg (1995) vorgestellte Beschreibungskonstrukt des Interaktionspunktes gibt zunächst nur den *rein formalen Charakter* der repräsentierten Formen wieder und lässt die Problematik der Semantik der einzelnen repräsentationalen Interaktionspunkte zunächst ausser acht (siehe in diesem Zusammenhang die Ergebnisse von Streit et al. 1989). Ein wesentlicher Vorteil unseres Konzeptes der Interaktionspunkte (FIPE) liegt jedoch darin, dass man neben dem 'Visualisierungsgrad' zusätzlich den 'interaktiven Aufwand' einfach beschreiben und quantifizieren kann. Wieviel verschiedene dialogfunktionale Interaktionspunkte (DFIPE) müssen z.B. benutzt werden, um an einen bestimmten anwendungsfunktionalen Interaktionspunkt (AFIP) heranzukommen? D.h., je weniger dialogfunktionale Interaktionspunkte notwendig sind, um einen benötigten AFIP zu erreichen, desto *direkter* ist die Interaktionsstruktur. Dieser Aspekt wird im Folgenden *interaktive Direktheit* genannt.

Allein jedoch mit den beiden Dimensionen *Visualisierungsgrad* (Gilmore 1991) und *interaktive Direktheit* (Laverson, Norman und Shneiderman 1987), können wir bereits eine brauchbare Klassifikation von einigen typischen Oberflächenarten vornehmen (siehe Abbildung 1). Interaktive Direktheit ist nicht identisch mit der Handlungsdirektheit von Laurel (1988); Handlungsdirektheit im Sinne von Laurel (1988) und Wandmacher (1993, S. 206f.) wird im Rahmen dieser Arbeit unter Aufgabenangemessenheit diskutiert.

Den Visualisierungsgrad kann man gemäss unserem Beschreibungskonzept einfach über das Verhältnis der repräsentationalen zu den funktionalen Interaktionspunkten bestimmen (siehe die Metrik "Feedback" in Rauterberg 1995). Wenn für alle vom interaktiven System angebotenen Funktionen mindestens eine spezifische Repräsentationsform existiert, so ist der Visualisierungsgrad  $\geq 100\%$ .

Je weniger Interaktionsschritte notwendig sind, um aus dem aktuellen Dialogkontext heraus an eine benötigte Anwendungsfunktion heranzukommen, desto *interaktiv direkter* ist die Oberfläche. Wenden wir diese

beiden Beschreibungsdimensionen auf die vier bekanntesten Oberflächenarten an, so zeigt sich, dass die Kommandooberfläche zwar genauso interaktiv direkt ist wie z.B. eine Desktopoberfläche (wenn nicht oftmals sogar direkter). Im Unterschied zu einer direktmanipulativen Oberfläche hat die Kommandooberfläche aber nur einen sehr geringen Visualisierungsgrad.

		Visualisierungsgrad	
		niedrig	hoch
interaktive Direktheit	niedrig	Stapelverarbeitung	(z.B. MsWORD für PCs) Menüoberfläche
	hoch	Kommando Oberfläche (z.B. MS-DOS, UNIX)	direkte Manipulation (z.B. Desktop-Oberfläche)

Abbildung 1 Klassifikationsschema für die verschiedenen Oberflächen mit den beiden Dimensionen Visualisierungsgrad und interaktive Direktheit. Wie man sehen kann, ist die Kommandooberfläche ähnlich interaktiv direkt wie eine direktmanipulierbare Oberfläche, hat jedoch einen ausserordentlich geringen Visualisierungsgrad.

## 3 Validierung der Klassifikation

Wie sich im letzten Abschnitt gezeigt hatte, unterscheiden sich die Menüoberfläche und die Oberfläche mit direkter Manipulation *nicht* hinsichtlich ihres Visualisierungsgrades. Um eine Validierung dieses Kriteriums ansatzweise zu ermöglichen, werden wir uns auf veröffentlichte Vergleichsstudien abstützen. Wenn wir gemäss unserem Klassifikationsschema annehmen, dass die Kommandooberfläche zwar einerseits eine maximale interaktive Direktheit aufweist, dafür aber im Vergleich zur Menüoberfläche aber nur einen minimalen Visualisierungsgrad hat, dann können wir folgendes daraus schliessen: Bei empirischen Vergleichsstudien zwischen Kommandooberfläche und Menüoberfläche wird sich *kein* klarer Vorteil für die eine oder andere Oberfläche ergeben können; jenachdem, ob der Visualisierungsgrad oder die interaktive Direktheit einen stärkeren Einfluss auf die aktuelle Interaktion hat, schneidet mal die Kommandooberfläche, mal die Menüoberfläche besser ab.

Wenn wir jedoch Kommandooberfläche mit direktmanipulativen Oberflächen vergleichen, dann sollte sich eine eindeutige Tendenz zugunsten der direktmanipulativen Oberflächen empirisch aufzeigen lassen. Eine direktmanipulative Oberfläche zeichnet sich nicht nur durch eine ähnlich hohe interaktive Direktheit wie eine Kommandooberfläche aus, sondern kann auch die Vorteile des hohen Visualisierungsgrades einer Menüoberfläche auf sich vereinen. Wir können daher

annehmen, dass die meisten veröffentlichten Vergleichsstudien einen Vorteil zugunsten der direktmanipulativen Oberfläche gegenüber der Kommando-oberfläche aufzeigen.

Zur Validierung des Visualisierungsgrades führen wir eine Meta-Analyse über bisher veröffentlichte empirische Vergleichsstudien durch (zum Thema Meta-Analyse siehe Hunter, Schmidt und Jackson 1982, sowie Rosenthal 1984). Wir haben alle Vergleichsstudien ausgewählt, in denen eine Kommando-oberfläche gegen eine Menü- oder eine Desktop-oberfläche getestet wurde. Dabei gehen wir davon aus, dass bei einer Menü- und einer Desktop-oberfläche eindeutig mehr visuelles Feedback vorhanden ist als bei einer Kommando-oberfläche.

### 3.1 Kommando- versus Menüoberfläche

Es lassen sich in der Literatur insgesamt neun Studien finden, in denen mindestens eine Kommando- mit einer Menüoberfläche verglichen wurde (siehe Tabelle 1). Die genaue Analyse der Ergebnisse aller neun Vergleichsstudien zeigt insgesamt ein eher uneinheitliches Bild bzgl. der Überlegenheit von Menüoberflächen über Kommando-oberflächen. Bei den 22 unterschiedlichen Messergebnissen schneidet die Menüoberfläche in neun Fällen besser ab (41%); in neun Fällen ergeben sich keine signifikanten Unterschiede (41%), und viermal zeigt sich sogar die Kommando-oberfläche der Menü-oberfläche signifikant überlegen (18%). Diese Vorteile einer Kommando-oberfläche zeigen sich tendenziell erst bei Fortgeschrittenen und Experten.

### 3.2 Kommando- versus Desktopoberfläche

Es lassen sich in der Literatur insgesamt 12 Studien finden, in denen mindestens eine Kommando- mit einer Desktop- oder direktmanipulierbaren Oberfläche verglichen wurde (siehe Tabelle 2).

Bei der Analyse der Ergebnisse dieser 12 Vergleichsstudien zeigt sich insgesamt eine deutliche Überlegenheit der Desktop-oberflächen über den Kommando-oberflächen. Bei 25 unterschiedlichen Messergebnissen schneidet die Desktop-oberfläche in 19 Fällen signifikant besser ab (76%); in fünf Fällen ergeben sich keine Unterschiede (20%), und nur einmal zeigt sich die Kommando-oberfläche der Desktop-oberfläche überlegen (4%).

## 4. Diskussion

Inwieweit die deskriptiven Ergebnisse im letzten Abschnitt auch einer inferenzstatistischen Überprüfung standhalten, wollen wir nun herausfinden. Dazu werden wir uns auf diejenigen Messergebnisse beschränken, bei denen sich ein signifikanter Unterschied zugunsten der einen oder anderen Oberfläche ergab.

Als erstes analysieren wir den Zusammenhang gemäss des von uns im Abschnitt 2 vorgestellten Klassifikationsschemas. Es ergibt sich eine Vierfeldertafel (siehe Tabelle 3), in der die empirisch gefundenen Häufigkeiten

für das Abschneiden im direkten Vergleich zwischen den drei Oberflächen eingetragen sind. Der signifikante  $\chi^2$ -Test für diese Vierfeldertafel ( $\chi^2 = 4.07$ ,  $df = 1$ ,  $p \leq .044$ ) lässt somit die Schlussfolgerung zu, dass eine KO im Vergleich zu einer DO signifikant öfters schlechter abschneidet als im Vergleich zu einer MO.

Tabelle 3: Kontingenztafel der Metaanalyse bzgl. des Oberflächenvergleichs anhand nur der signifikanten Unterschiede in Tabelle 1 und 2 [Zelleninhalt: beobachtete (erwartete) Häufigkeit]

	MO	DO
KO besser als...	4 (2.0)	1 (3.0)
KO schlechter als...	9 (11.0)	19 (17.0)

Wenn dem so ist, wie kann es dann noch Benutzergruppen geben, die nach wie vor eine KO bevorzugen? Diese Benutzergruppe setzt sich oftmals aus Personen mit ausreichender Computererfahrung zusammen. Versuchen wir also die oft gestellte Frage zu beantworten: Sind KO für Experten geeignet?

Dazu haben wir in den Tabellen 1 und 2 alle Testpersonen mit wenig Computererfahrung (Anfänger) der Gruppe mit viel Erfahrung gegenübergestellt (Fortgeschrittene + Experten). Dann haben wir ausgezählt, wie oft eine KO besser bzw. schlechter als eine MO oder DO abgeschnitten hat (siehe Tabelle 4). Der signifikante  $\chi^2$ -Test für diese Vierfeldertafel ( $\chi^2 = 5.55$ ,  $df = 1$ ,  $p \leq .018$ ) lässt somit die Schlussfolgerung zu, dass Benutzer mit entsprechend grosser Computererfahrung eher von einer KO profitieren.

Tabelle 4: Kontingenztafel der Metaanalyse bzgl. der Vorerfahrung anhand nur der signifikanten Unterschiede in Tabelle 1 und 2 [Zelleninhalt: beobachtete (erwartete) Häufigkeit]

	Anfänger	Fortge.
KO besser als MO oder DO	0 (2.4)	5 (2.6)
KO schlechter als MO oder DO	16 (13.6)	12 (14.4)

Dass jedoch der Umstieg auf eine vollgraphische Desktop-oberfläche alleine – ohne weitere Gestaltungsüberlegungen (siehe z.B. Marais 1990, S. 13, bzw. Olsen und Holladay 1994) – nicht ausreicht, konnte mit einer Vergleichsstudie verschiedener graphischer Oberflächen gezeigt werden (PC-Professionell, 1995).

*Tabelle 1 Übersicht über neun verschiedene Vergleichsstudien: KO vs. MO (KO = Kommandooberfläche, MO = Menüoberfläche, DO = Desktopoberfläche, IO = 'ikonenorientierte' Oberfläche, NO = natürlichsprachliche Oberfläche; 'O<sub>x</sub> < O<sub>y</sub>' = O<sub>x</sub> ist schlechter als O<sub>y</sub>, 'O<sub>x</sub> > O<sub>y</sub>' = O<sub>x</sub> ist besser als O<sub>y</sub>; "sig." bedeutet, dass das Signifikanzniveau  $p \leq 0.05$  ist, "nicht sig." bedeutet, dass  $p > 0.05$  ist).*

Literaturstelle	Oberfläche(n)	Testperson	Mess-Skala	Ergebnis	Signif.
Streitz et al. (1987)	KO, MO	Anfänger	Bearbeitungszeit	KO < MO	sig.
Chin et al. (1988)	KO, MO	Anfänger	subjektive Bewertung	KO < MO	sig.
Ogden & Boyle (1982)	KO, MO, HY	Anfänger	Präferenz	KO < MO	sig.
Roy (1992)	KO, MO	Fortgeschritten	Fehlerrate	KO < MO	sig.
Roberts & Moran (1983)	KO, MO, DO	Experte	Bearbeitungszeit	KO < MO	sig.
Chin et al. (1988)	KO, MO	Experte	subjektive Bewertung	KO < MO	sig.
Peters et al. (1990)	KO, MO, DO	Experte	Anzahl Slips	KO < MO	sig.
Peters et al. (1990)	KO, MO, DO	Experte	Erinnerungsfehler	KO < MO	sig.
Peters et al. (1990)	KO, MO, DO	Experte	Effizienz	KO < MO	sig.
Ogden & Boyle (1982)	KO, MO, HY	Anfänger	Bearbeitungszeit	KO < MO	nicht sig.
Roy (1992)	KO, MO	Fortgeschritten	Bearbeitungszeit	KO < MO	nicht sig.
Antin (1988)	KO, MO, KMO	Fortgeschritten	subjektive Bewertung	KO < MO	nicht sig.
Hauptmann & Green (1983)	KO, MO, NO	Anfänger	Bearbeitungszeit	KO = MO	nicht sig.
Hauptmann & Green (1983)	KO, MO, NO	Anfänger	Anzahl Fehler	KO = MO	nicht sig.
Hauptmann & Green (1983)	KO, MO, NO	Anfänger	subjektive Bewertung	KO = MO	nicht sig.
Whiteside et al. (1985)	KO, MO, IO	Anfänger	Lösungsgüte	KO > MO	nicht sig.
Antin (1988)	KO, MO, KMO	Fortgeschritten	Präferenz	KO > MO	nicht sig.
Roberts & Moran (1983)	KO, MO, DO	Experte	fehlerfreie Bearbeitungszeit	KO > MO	nicht sig.
Whiteside et al. (1985)	KO, MO, IO	Fortgeschritten	Lösungsgüte	KO > MO	sig.
Streitz et al. (1987)	KO, MO	Fortgeschritten	Bearbeitungszeit	KO > MO	sig.
Antin (1988)	KO, MO, KMO	Fortgeschritten	Lösungsgüte	KO > MO	sig.
Whiteside et al. (1985)	KO, MO, IO	Experte	Lösungsgüte	KO > MO	sig.

*Tabelle 2 Übersicht über 12 verschiedene Vergleichsstudien: KO vs. DO (KO = Kommandooberfläche, MO = Menüoberfläche, DO = Desktopoberfläche; 'O<sub>x</sub> < O<sub>y</sub>' = O<sub>x</sub> ist schlechter als O<sub>y</sub>, 'O<sub>x</sub> > O<sub>y</sub>' = O<sub>x</sub> ist besser als O<sub>y</sub>).*

Literaturstelle	Oberflächen	Testpersonen	Mess-Skala	Ergebnis	Signif.
Altmann (1987)	KO, DO	Anfänger	Bearbeitungszeit	KO < DO	sig.
Karat et al. (1987)	KO, DO	Anfänger	Bearbeitungszeit	KO < DO	sig.
Streitz et al. (1989)	KO, DO	Anfänger	Bearbeitungszeit	KO < DO	sig.
Sengupta & Te'eni (1991)	KO, DO	Anfänger	Bearbeitungszeit	KO < DO	sig.
Margono et al. (1987)	KO, DO	Anfänger	Anzahl Fehler	KO < DO	sig.
Morgan et al. (1991)	KO, DO	Anfänger	Anzahl Fehler	KO < DO	sig.
Morgan et al. (1991)	KO, DO	Anfänger	Zeit zwischen Fehlern	KO < DO	sig.
Karat et al. (1987)	KO, DO	Anfänger	Fehlerkorrekturzeit	KO < DO	sig.
Morgan et al. (1991)	KO, DO	Anfänger	fehlerfreie Zeit	KO < DO	sig.
Margono et al. (1987)	KO, DO	Anfänger	subjektive Bewertung	KO < DO	sig.
Morgan et al. (1991)	KO, DO	Anfänger	subjektive Bewertung	KO < DO	sig.
Torres-Chazaro et al.(1992)	KO, DO	Anfänger	subjektive Bewertung	KO < DO	sig.
Sengupta & Te'eni (1991)	KO, DO	Anfänger	effektive Nutzung	KO < DO	sig.
Tombaugh et al. (1989)	KO, DO	Fortgeschritten	subjektive Bewertung	KO < DO	sig.
Torres-Chazaro et al.(1992)	KO, DO	Fortgeschritten	subjektive Bewertung	KO < DO	sig.
Roberts & Moran (1983)	KO, MO, DO	Experte	Bearbeitungszeit	KO < DO	sig.
Peters et al. (1990)	KO, MO, DO	Experte	Flüchtigkeitsfehler	KO < DO	sig.
Peters et al. (1990)	KO, MO, DO	Experte	Erinnerungsfehler	KO < DO	sig.
Peters et al. (1990)	KO, MO, DO	Experte	Effizienz	KO < DO	sig.
Margono et al. (1987)	KO, DO	Anfänger	Bearbeitungszeit	KO < DO	nicht sig.
Morgan et al. (1991)	KO, DO	Anfänger	Bearbeitungszeit	KO < DO	nicht sig.
Tombaugh et al. (1989)	KO, DO	Fortgeschritten	Bearbeitungszeit	KO < DO	nicht sig.
Roberts & Moran (1983)	KO, MO, DO	Experte	Fehlerkorrekturzeit	KO < DO	nicht sig.
Altmann (1987)	KO, DO	Anfänger	subjektive Bewertung	KO > DO	nicht sig.
Masson et al. (1988)	KO, DO	Fortgeschritten	Bearbeitungszeit	KO > DO	sig.

## 5. Zusammenfassung

Ausgangspunkt unserer Meta-Analyse war die Annahme, dass ein hoher Visualisierungsgrad gepaart mit grosser interaktiver Direktheit von Vorteil ist. Während wir beim Vergleich von Menüoberflächen mit Kommando-oberflächen ein – wie erwartet – eher uneinheitliches Bild bekommen (siehe Tabelle 1), so zeigt sich die Überlegenheit von Desktop- bzw. direktmanipulativen Oberflächen deutlich (siehe Tabelle 2). Die uneinheitlichen Ergebnisse bei den Menüoberflächen können zum Teil durch den Trade-off zwischen hohem 'Visualisierungsgrad' und geringer 'interaktiver Direktheit' erklärt werden (siehe auch Abbildung 1). Die wahrscheinlich zu geringe interaktive Direktheit der Menüoberflächen verhindert dann einen eventuell vorhandenen Vorteil eines hohen Visualisierungsgrades. Die Richtigkeit dieser Erklärung kann durch die eindeutigen Vergleichsergebnisse zugunsten der direktmanipulierbaren Desktopoberflächen im Rahmen einer statistischen Metaanalyse belegt werden.

Zusammenfassend lässt sich festhalten, dass – bei vergleichbarer 'interaktiver Direktheit' – das Ausmass an Feedback über das aktuell gültige Operatorsystem (der Visualisierungsgrad) von entscheidender Bedeutung ist. Ob und inwieweit jedoch allein das Ausmass an visuellem Feedback und an interaktiver Direktheit für einen empirisch beobachtbaren Vorteil verantwortlich ist, bleibt zunächst dahingestellt (siehe Rauterberg 1995). Insgesamt betrachtet, können wir die hier vorgestellten Ergebnisse mit der entsprechenden Interpretation als eine Bestätigung dafür ansehen, dass unser quantitatives Beschreibungskonzept erfolgreich als Grundlage für eine weitere theoretische Durchdringung softwareergonomischer Gestaltungsfragen eingesetzt werden kann.

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

## 8. Literaturverzeichnis

Altmann, A. (1987) Direkte Manipulation: empirische Befunde zum Einfluss der Benutzeroberfläche auf die Erlernbarkeit von Textsystemen. *Zeitschrift für Arbeits- und Organisationspsychologie* 31(3):108-114.

Antin, J. (1988) An empirical comparison of menu selection, command entry and combined modes of computer control. *Behaviour and Information Technology* 7(2):173-182.

Balzert, H. (1988) E/A-Geräte für die Mensch-Computer-Interaktion. In: H. Balzert, H. Hoppe, R. Oppermann, H. Peschke, G. Rohr & N. Streitz (Hrsg.) *Einführung in die Software-Ergonomie. (Mensch-Computer Kommunikation – Grundwissen 1, S. 67-98)*. Berlin: de Gruyter.

Bannert, M. (1991) Benutzerfreundlichkeit von direktmanipulativen Schnittstellen: Reihenfolge der Objekt- und Funk-

tionsspezifikation und expliziter versus impliziter Funktionsaufruf. unveröffentlichte Diplomarbeit, (Psychologisches Institut), Heidelberg: Ruprecht-Karls-Universität.

Chin, J. P., Diehl, V. A. & Norman, D. (1988) Development of an instrument measuring user satisfaction of the human-computer interface. In: E. Soloway, D. Frye & S. B. Sheppard (eds.) *Human Factors in Computing Systems - CHI'88*. New York: ACM, pp. 213-218.

Dannenberg, R. & Blattner, M. (1992) Introduction: the trends toward multimedia interfaces. In: M. Blattner & R. Dannenberg (eds.) *Multimedia Interface Design*. New York: ACM Press, pp. xvii-xxv.

Dumais, S. T. & Jones, W. P. (1985) A Comparison of Symbolic and Spatial Filing. In: L. Borman & B. Curtis (eds.) *Human Factors in Computing Systems-II*. Amsterdam: Elsevier, pp. 127-130.

Fabian, F. & Rathke, C. (1983) Menüs: Einsatzmöglichkeiten eines Fenstersystems zur Unterstützung der Mensch-Maschine-Kommunikation. *Office Management* 31(Sonderheft): 42-44.

Foley, J. D. (1988) Neuartige Schnittstellen zwischen Mensch und Computer. *Spektrum der Wissenschaft, Sonderheft* Nr. 6 "Die nächste Computer-Revolution", S. 66-69.

Fujita, T., Itoh, H., Taguchi, H., Fukuoka, T., Nishiyama, S. & Watanabe, K. (1993) Simulated world of hypothetical life forms – virtual creatures. In: G. Salvendy & M. J. Smith (eds.) *Human-Computer Interaction: Software and Hardware Interfaces. (Advances in Human Factors/-Ergonomics 19B, pp. 663-666)* Amsterdam: Elsevier.

Gilmore, D. (1991) Visibility: a dimensional analysis. In: D. Diaper & N. Hammond (eds.) *People and Computers VI*. Cambridge: Cambridge Univ. Press, pp. 317-329.

Greenberg, S. & Witten, I. H. (1988) Directing The User Interface: How People Use Command-Based Computer Systems. Preprints of the IFAC/IFIP/IEA/ IFORS Conference on Man-Machine Systems: Analysis, Design and Evaluation, vol. II, pp 299-305.

Hächler, G., Julen, M., Keller, R. & Stoffel, M. (1990) Arbeitspsychologische Analyse moderner Benutzungsoberflächen der MCI. unveröffentlichte Semesterarbeit, Institut für Arbeitspsychologie, Zürich: ETH.

Hauptmann, A. G. & Green, B. F. (1983) A comparison of command, menu-selection and natural-language computer programs. *Behaviour and Information Technology* 2(2): 163-178.

Hunter, J., Schmidt, F. & Jackson, G. (1982) *Meta-Analysis: cumulating research findings across studies*. London: Sage.

Ilg, R. & Ziegler, J. (1987) Interaktionstechniken. In: K-P. Fähnrich (Hrsg.) *Software-Ergonomie*. München: Oldenbourg, State of the Art, No. 5., S 106-117.

Karat, J., Fowler, R. & Gravelle, M. (1987) Evaluating user interface complexity. In: H-J. Bullinger & B. Shackel (eds.) *Human-Computer Interaction - INTERACT '87*. Amsterdam: North-Holland, pp. 489-495.



- Koller, F. (1992) Gestaltung von Multimedia Systemen. *Ergonomie & Informatik* 17:3-7.
- Koller, F. & Ziegler, J. (1989) Benutzerpräferenzen bei alternativen Eingabetechniken. In: S. Maass & H. Oberquelle (Hrsg.) *Software-Ergonomie '89*. (German Chapter of the ACM Berichte, Band 32, S. 304-312), Stuttgart: Teubner.
- König, A. (1989) Desktop als Mensch-Maschine-Schnittstelle. (Springers *Angewandte Informatik*, H. Schauer, Hrsg.), Wien: Springer.
- Kraut, R E., Hanson, S. J., Farber, J. M. (1983) Command Usage and Interface Design. In: A. Janda (ed.) *Human Factors in Computing Systems - CHI'83*. Amsterdam: Elsevier, pp 120-124.
- Laurel, B. (1988) Interface as mimesis. In: D.A. Norman & S.W. Draper (eds.) *User centered system design*. Hillsdale: Erlbaum, pp 67-85.
- Lauter, B. (1987) *Software-Ergonomie in der Praxis*. München: Oldenbourg.
- Laverson, A., Norman, K. & Shneiderman, B. (1987) An evaluation of jump-ahead technique in menu selection. *Behaviour and Information Technology* 6(2), 97-108.
- Marais, J. (1990) The GADGETS User Interface Management System. Technical Report No. 144, Departement Informatik, Zürich: Eidgenössische Technische Hochschule.
- Margono, S. & Shneiderman, B. (1987) A study of file manipulation by novices using commands vs. direct manipulation. In: *Proceedings of 26th Annual Technical Symposium*, Washington D.C. Chapter of ACM Gaithersburg, MD - June, 11, 1987. New York: ACM, pp. 154-159.
- Mel, B. W., Omohundor, S. M., Robinson, A. D., Skiena, S. S., Thearling, K. H., Young, L. T., Wolfram, S. (1988) *Tablet: Personal Computer in the Year 2000*. *Communications of the ACM* 31(6):639-646.
- Morgan, K., Morris, R. & Gibbs, S. (1991) When does a mouse become a rat? or ... comparing performance and preferences in direct manipulation and command line environment. *The Computer Journal* 34(3):265-271.
- Nielsen, J. (1993) *Usability Engineering*. London: Academic Press.
- Norman, D. A. (1986) *Cognitive Engineering*. In: D.A. Norman & S.W. Draper (eds.) *User centered system design*. Hillsdale: Erlbaum, pp 31-61.
- Oberquelle, H. (1994) Formen der Mensch-Computer-Interaktion. In: E. Eberleh, H. Oberquelle & R. Oppermann (Hrsg.) *Einführung in die Software-Ergonomie*. (2. Auflage), Berlin: de Gruyter, S.95-143.
- Olsen, D. & Holladay, W. (1994) Automatic generation of interactively consistent search dialogs. In: B. Adelson, S. Dumais & J. Olson (eds.) *Human Factors in Computing Systems - CHI'94*. New York: ACM, pp. 218-224.
- PC-Professionell (1995) Es muss nicht immer Windows sein. *PC Professionell* (Februar): 68-118.
- Peters, H., Frese, M. & Zapf, D. (1990) Funktions- und Nutzungsprobleme bei unterschiedlichen Dialogformen. *Zeitschrift für Arbeitswissenschaft* 44(3):145-152.
- Rauterberg, M. (1991) Benutzungorientierte Benchmark-Tests: eine Methode zur Benutzerbeteiligung bei Standardsoftwareentwicklungen. (Projektbericht zum Forschungsprojekt "Benutzerorientierte Softwareentwicklung und Schnittstellengestaltung BOSS", Nr. 6, P. Spinass, M. Rauterberg, O. Strohm, D. Waeber & E. Ulich, Hrsg.) Zürich: Institut für Arbeitspsychologie.
- Rauterberg, M. (1993) A product oriented approach to quantify usability attributes and the interactive quality of user interfaces. In: H. Luczak, A. Cakir & G. Cakir (Eds.) *Work With Display Units 92*. Amsterdam: North-Holland, pp. 324-328.
- Rauterberg, M. (1995) Über die Quantifizierung softwareergonomischer Richtlinien. *Ergonomie & Informatik* 25:5-18.
- Roberts, T. & Moran, T. (1983) The Evaluation of Text Editors: Methodology and Empirical Results. *Communications of the ACM* 26(4):265-283.
- Rosenthal, R. (1984) Meta-analysis procedures for social research. (*Applied Social Research Methods Series*, Vol. 6), London: Sage.
- Roy, G. (1992) An evaluation of command line and menu interface in a CAD environment. *International Journal of Computer Integrated Manufacturing* 5(2):94-106.
- Sengupta, K. & Te'eni D. (1991) Direct manipulation and command language interfaces: a comparison of users' mental models. In: H.-J. Bullinger (ed.) *Human Aspects in Computing: Design and Use for Interactive Systems and Information Management*. (*Advances in Human Factors/Ergonomics*, 18A, pp. 429-434), Amsterdam : Elsevier.
- Shneiderman, B. (1983) Direct manipulation: a step beyond programming languages. *IEEE Computer* 16(8):57-69.
- Shneiderman B (1987) *Designing the User Interface*. Addison-Wesley, Reading MA.
- Smith, D. C., Irby, C., Kimball, R., Verplank, B. & Harslem, E. (1982) *Designing the Star User Interface*. *Byte* 7(4):242-282.
- Sommerville, I. (1987) *Software Engineering*. 2nd edition. Amsterdam: Addison-Wesley.
- Staufer, M. (1987) *Piktogramme für Computer*. Berlin: de Gruyter.
- Streitz, N., Spijkers, W. A. C. & van Duren, L. L. (1987) From Novice to expert user: a transfer of learning on different interaction modes. In: H.-J. Bullinger & B. Shackel (eds.) *Human-Computer Interaction - INTERACT '87*. New York: North-Holland, pp. 841-846.
- Streitz, N., Lieser, A. & Wolters, A. (1989) The combined effects of metaphor worlds and dialogue modes in human-computer-interaction. In: F. Klix, N. Streitz, Y. Waern & H. Wandke (eds.) *Man-Computer Interaction Research MACINTER-II*. Amsterdam: Elsevier, pp. 75-88.
- Tombaugh, J., Paynter, B. & Dillon, R. (1989) Command and graphic interfaces: user performance and satisfaction. In: G. Salvendy & M. Smith (eds.) *Designing and Using Hu-*

man-Computer Interfaces and Knowledge Based Systems. Amsterdam: Elsevier, pp. 369-375.

Torres-Chazaro, O., Beaton, R. & Deisenroth, M. (1992) Comparison of command language and direct manipulation interfaces for CNC milling machines. *International Journal of Computer Integrated Manufacturing* 5(2):107-117.

Ulich, E., Rauterberg, M., Moll, T., Greutmann, T. & Strohm, O. (1991): Task orientation and User-Oriented Dialog Design. *International Journal of Human-Computer Interaction* 3(2), 117-144.

Vossen, P., Wandel, B., Morlock, P., Federmann, B. & Thiermeyer, S. (1991) Spracheingabe an einem CAD-System: ein empirischer Versuch unter Verwendung der "Wizard of Oz" Technik. *Ergonomie & Informatik* 14:7-13.

Wandmacher, J. (1993) Software-Ergonomie. (Mensch-Computer-Kommunikation: Grundwissen, Band 2; H. Balzert, Hrsg.), Berlin: de Gruyter.

Ward, J. R. & Phillips, M. J. (1987) Digitizer Technology: Performance Characteristics and the Effects on the User Interface. *IEEE Computer Graphics and Applications* 4:31-44.

Wellner, P. (1993) Interacting with Paper on the DigitalDesk. *Communications of the ACM* 36(7):86-96.

Whiteside, J., Jones, S., Levy, P. S. & Wixon, D. (1985) User Performance with Command, Menu, and Iconic Interfaces. *Human Factors in Computing Systems-II. Proceedings of the CHI '85 Conference in San Francisco, Amsterdam New York Oxford: North-Holland*, pp. 185-191.

Zeidler, A. & Zellner, R. (1992) *Software-Ergonomie: Techniken der Dialoggestaltung*. München: Oldenbourg.

Ziegler, J. (1987) Grunddimensionen von Interaktionsformen. In: W. Schönplflug & M. Wittstock (Hrsg.) *Software-Ergonomie '87*. (Berichte des German Chapter of the ACM, Band 29, S. 489-497), Stuttgart: Teubner.

Ziegler, J. (1993) Entwurf graphischer Benutzungsschnittstellen. In: J. Ziegler & R. Ilg (Hrsg.) *Benutzergerechte Software-Gestaltung*. München: Oldenbourg, S. 145-169.

Zue, V., Glass, J., Goddeau, D., Goodine, D., Hirschman, L., Leung, H., Phillips, M., Polifroni, J. & Seneff, S. (1991) Spoken language systems for human/machine interfaces. In: *Conference Proceedings RIAO'91 "Intelligent Text and Image Handling"* Vol. 2, held in Barcelona (Sp), April 2-5, 1991, pp. 936-955.