

Literaturarbeit der Philosophischen Fakultät I
der Universität Zürich

Entwicklung der Menüforschung in Human Computer Interaction

März 2002

Christian Hübscher
Stettbacherrain 1
8051 Zürich

Psychologisches Institut der Universität Zürich
Abteilung Angewandte Psychologie
Fachrichtung Psychologische Methodenlehre

Prof. R. Hirsig
Betreuung: Dr. Th. E. Rothenfluh

Zusammenfassung

Das Menü-Konzept ist schon seit längerem bekannt und ist auch aus modernen User Interfaces nicht mehr wegzudenken. Aus diesem Grund hat die Menüforschung eine lange Tradition, und das Thema ist auch heute immer noch Gegenstand der Forschung. Die vorliegende Arbeit soll diese Entwicklung im Gebiet Human Computer Interaction aufzeigen.

Ein Schwerpunkt des interdisziplinären Gebietes Human Computer Interaction (HCI) ist die Gestaltung benutzerfreundlicher User Interfaces von Computersystemen. Wichtige User Interface Paradigmen sind Kommandozeile, Menübasiertes Interface, Direkt Manipulation / Graphical User Interface (GUI) sowie Web Interface. Ausser bei der Kommandozeile werden bei den anderen Interaktionsstilen Menüs eingesetzt, die jedoch verschiedene Eigenschaften haben.

Die eher klassischen Untersuchungen zu Menüs befassen sich vor allem mit deren Strukturierung. In Bezug auf die Struktur über mehrere Menüs wurde vor allem der «depth-breadth trade-off» ausführlich untersucht. Die Frage nach der Struktur innerhalb eines Menüs hat ebenfalls zu vielfältiger Forschung geführt. Das Fitts's Law war die Grundlage für verschiedene Optimierungen von Menüs in GUIs und neuen Ansätzen wie Pie Menüs.

In der neueren Forschung wurde die Frage nach der Suche einzelner Einträge in einem Menü wieder aufgenommen und mittels kognitiver Modellierung oder Eyetracking untersucht. Für das Problem von Menüs mit sehr vielen Menüoptionen wurden verschiedene konkrete Lösungen wie Split Menüs oder Fisheye Menüs vorgeschlagen. Aus dem aktiven HCI Forschungsgebiet der Information Visualization wurden verschiedene Techniken zur Darstellung und Navigation grosser Datenmengen vorgestellt. Mit dem Einsatz von Transparenz wurde ein Mittel vorgestellt, den Arbeitskontext der Benutzer möglichst zu erhalten und die Marking Menüs entwickeln das Konzept der Pie Menüs weiter.

Mit der professionellen 3-D Animationssoftware Maya wird ein erfolgreiches kommerzielles Produkt vorgestellt, welches fortschrittliche Menükonzepte wie Marking Menüs und Transparenz einsetzt. Die Hotbox ist ein Konzept, welches speziell für die konkreten Anforderungen der Benutzer von Maya entwickelt wurde.

Zwar scheinen zu einigen Fragen der Strukturierung und Präsentation von Menüs die Forschungsbemühungen abgeschlossen zu sein. In beiden Bereichen ergeben sich durch die Entwicklung aber immer wieder neue Fragen. So ist auch ein Ende der Menüforschung nicht abzusehen.

Inhaltsverzeichnis

1. Einleitung.....	5
2. Theoretischer Rahmen	6
2.1. Human Computer Interaction (HCI).....	6
2.2. Usability (Benutzerfreundlichkeit)	7
2.3. User Interface	8
2.3.1. Begriffsdefinition	8
2.3.2. Verschiedene User Interface Paradigmen – Interaktionsstile	8
2.3.2.1. Kommandozeile	9
2.3.2.2. Menübasiertes Interface	9
2.3.2.3. Direktmanipulation / Graphical User Interface (GUI).....	11
2.3.2.4. Web Interface	15
3. Klassische Fragen der Menüforschung.....	17
3.1. Strukturierung von Menüs.....	17
3.1.1. Struktur über mehrere Menüs	17
3.1.2. Struktur innerhalb eines Menüs.....	18
3.2. Präsentation von Menüs.....	18
3.2.1. Fitts's Law	19
3.2.2. Optimierung von Menüs	19
4. Neuere Menüforschung.....	23
4.1. Aktuelle Themen der Forschung	23
4.1.1. Suche innerhalb von Menüs	23
4.1.2. Menüs für eine grosse Menge von Optionen	24
4.1.2.1. Split Menüs	24
4.1.2.2. Fisheye Menüs	26
4.1.2.3. Ansätze aus dem Gebiet Information Visualization	26
4.1.3. Erhaltung des Kontextes durch den Einsatz von Transparenz	28
4.1.4. Marking Menüs – eine Weiterentwicklung der Pie Menüs	29
4.2. «Menüs der Zukunft» in der Praxis – Maya von Alias Wavefront.....	30
4.2.1. Marking Menüs.....	31
4.2.2. Die Hotbox	32
5. Diskussion	34
5.1. Strukturierung von Menüs.....	34
5.2. Präsentation von Menüs	35
Literaturverzeichnis.....	36

Verzeichnis der Abbildungen

Abbildung 1 Comic (von www.userfriendly.org)	5
Abbildung 2 Aspekte von HCI (aus SIGCHI, 2001)	6
Abbildung 3 Komponenten eines Mensch-Maschine-Systems (angelehnt an Shackel, 1991, S. 23)	7
Abbildung 4 DOS Kommandozeile (Screenshot aus Windows 98)	9
Abbildung 5 Menübasierte Anwendung von IBM (aus Mandel, 1997).....	10
Abbildung 6 Menü eines Mobiltelefons (von www.nokia.com)	11
Abbildung 7 Beispiel von Direktmanipulation (Screenshot aus Illustrator)	12
Abbildung 8 GUI von Xerox Star (aus Johnson et al., 1995)	12
Abbildung 9 Menü «Bearbeiten» in der Menüleiste (Screenshot aus Mac OS).....	13
Abbildung 10 Pop-up Menü in einer Dialogbox (Screenshot aus Mac OS)	14
Abbildung 11 Kontextmenüs (Screenshot aus Mac OS).....	14
Abbildung 12 Beispiele von Menüs auf www.cnn.com und www.yahoo.com	15
Abbildung 13 Beispiele von Menüs auf www.apple.com und www.amazon.de	16
Abbildung 14 Ordnungsschemata für Menüs (aus Mayhew, 1992)	18
Abbildung 15 Fitts's Law schematisch (nach Raskin, 2000)	19
Abbildung 16 Menüleiste von Mac OS (Screenshot aus Mac OS)	20
Abbildung 17 Menüleiste aus Windows'98 Applikation (Screenshot aus Windows'98)	20
Abbildung 18 Hierarchisches Pull-down Menü (Screenshot aus Mac OS).....	21
Abbildung 19 Auswahl eines Menüpunktes (Screenshot aus Mac OS).....	21
Abbildung 20 Pie Menü (aus Shneiderman, 1998).....	22
Abbildung 21 Split Menüs (aus Sears und Shneiderman,1994 und Screenshot aus Word).....	24
Abbildung 22 Fisheye Menü (von www.cs.umd.edu/hcil).....	25
Abbildung 23 Cone Tree (von www.inxight.com)	26
Abbildung 24 Table Lens (aus Rao und Card, 1994)	27
Abbildung 25 Treemap schematisch (aus Herman et al., 2000).....	27
Abbildung 26 LifeLines (aus Plaisant et al., 1996)	28
Abbildung 27 Click-trough Buttons (aus Bier et al., 1993).....	28
Abbildung 28 Transparente Menüs mit verschiedenen Schriften (aus Harrison und Vicente, 1996).....	29
Abbildung 29 Marking Menü (aus Kurtenbach und Buxton, 1993)	30
Abbildung 30 Maya User Interface (von www.aliaswavefront.com).....	30
Abbildung 31 Objekt mit Marking Menü (von www.aliaswavefront.com)	31
Abbildung 32 Maya Hotbox (aus Kurtenbach et al., 1999)	32
Abbildung 33 Lineares Menü in der Hotbox (aus Kurtenbach et al., 1999)	32
Abbildung 34 Marking Menü in der Hotbox (aus Kurtenbach et al., 1999)	33

Screenshots wurden erstellt aus: Apple Mac OS 9.1, Microsoft Windows 98 (auf Virtual PC 4.0 für Macintosh), Microsoft Word 98 für Macintosh und Adobe Illustrator 7.0 für Macintosh.

1. Einleitung

Jeder Computerbenutzer¹ kennt Menüs in irgend einer Form. Auch viele andere Geräte werden über Menüs bedient: Mobiltelefone, Geldautomaten, Waschmaschinen etc. Menüs in technischen Geräten sind nicht mehr wegzudenken. Man könnte beinahe vergessen, dass sie ursprünglich eine Errungenschaft der Gastronomen und nicht der Ingenieure sind, wie Abbildung 1 zeigt.

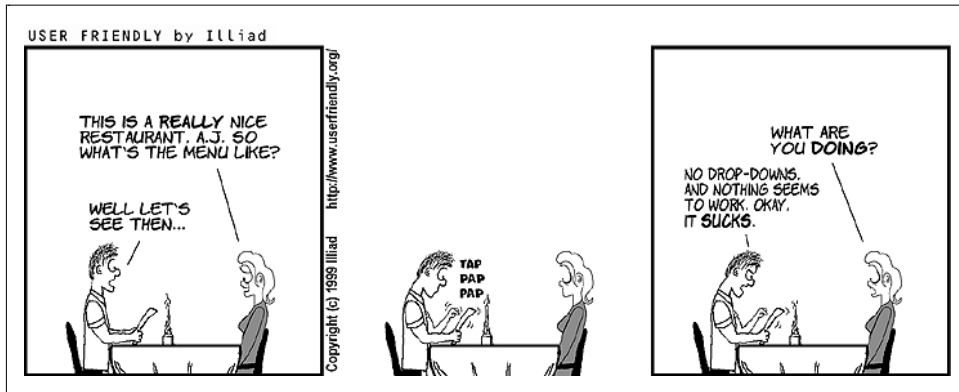


Abbildung 1 Comic (von www.userfriendly.org)

Schon lange beinhalten User Interfaces von Computern Menüs, und auch in der heutigen Zeit werden sie mit Erfolg eingesetzt. Dabei gibt es unzählige Gestaltungsformen von Menüs. Aber auch die Forschung zur Gestaltung von Menüs hat im Gebiet Human Computer Interaction eine lange Tradition. In unzähligen Experimenten wurden verschiedene Arten von Menüs miteinander verglichen und immer wieder werden neue Umsetzungen des Menüprinzips entwickelt.

In dieser Arbeit soll die Entwicklung der Menüforschung im Gebiet Human Computer Interaction aufgezeigt werden. Die Unterscheidung in «klassische» und «neuere Menüforschung» ist dabei eher willkürlich und soll grob unterscheiden in Erkenntnisse, die bereits in grösserem Umfang in Produkten umgesetzt wurden oder welchen, die nur in der Forschung oder unter Benutzern von spezialisierten Programmen bekannt sind.

¹ Geschlechtsspezifische Begriffe: Zur Vereinfachung und somit besseren Lesbarkeit des Textes werden in dieser Arbeit maskuline Wortformen verwendet. So ist z.B. mit «Benutzer» eine Person gemeint, die etwas benutzt, unabhängig vom Geschlecht.

2. Theoretischer Rahmen

Ein Schwerpunkt des interdisziplinären Gebietes Human Computer Interaction ist die Gestaltung benutzerfreundlicher User Interfaces von Computersystemen. Wichtige User Interface Paradigmen sind Kommandozeile, Menübasiertes Interface, Direkt Manipulation / Graphical User Interface sowie Web Interface. Ausser bei Kommandozeilen werden bei den anderen Interaktionsstilen Menüs eingesetzt, die jedoch verschiedene Eigenschaften haben.

2.1. Human Computer Interaction (HCI)

Für das Feld Human Computer Interaction (HCI) besteht keine allgemein akzeptierte Definition. Die Arbeitsdefinition der SIGCHI – ACM Special Interest Group on Computer-Human Interaction (2001) für die Entwicklung eines HCI-Curriculums lautet:

Human-computer interaction is a discipline concerned with the design, evaluation and implementation of interactive computing systems for human use and with the study of major phenomena surrounding them.

Das interdisziplinäre Gebiet entwickelt sich als Spezialgebiet innerhalb verschiedener Disziplinen wie Informatik, Psychologie, Soziologie und Visual Design mit ihren eigenen Schwerpunkten. Abbildung 2 zeigt einen Überblick über die behandelten Bereiche.

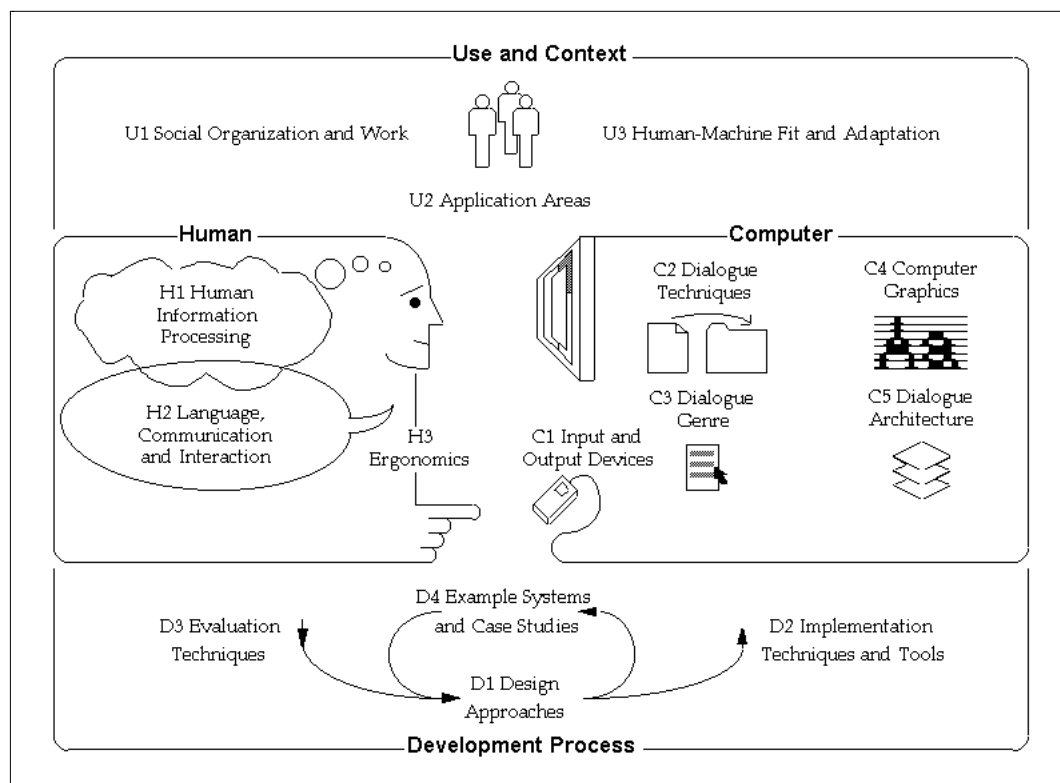


Abbildung 2 Aspekte von HCI (aus SIGCHI, 2001)

Die Arbeit mit einem Computer d.h. eine Mensch-Maschine-Situation ist nach Shackel (1991, S. 24) immer charakterisiert durch den *Benutzer*, der in einem bestimmten *Umfeld* mit einem *Werkzeug* – hier mit einem Computer – eine bestimmte *Aufgabe* erfüllt (Abbildung 3).

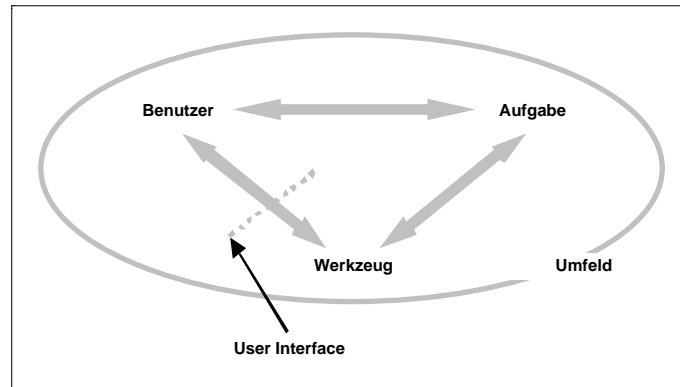


Abbildung 3 Komponenten eines Mensch-Maschine-Systems (angelehnt an Shackel, 1991, S. 23)

Die Benutzer interagieren mit dem Computer über das User Interface. Eines der Hauptanliegen des Gebietes Human Computer Interaction ist es, ein User Interface so zu gestalten, dass es eine möglichst hohe Usability aufweist.

2.2. Usability (Benutzerfreundlichkeit)

Die Usability eines Systems ist nach Shackel (1991) wie folgt definiert:

... the usability of a system or equipment is: 'the capability in human functional terms to be used easily and effectively by the specified range of users, given specified training and user support, to fulfil the specified range of tasks, within the specified range of environmental scenarios'. (S. 24)

Kann ein System so gestaltet werden, dass es möglichst gut auf die Benutzer, die Aufgabe und das Umfeld abgestimmt ist, zeichnet es sich durch hohe Usability aus.

Usability ist nicht eine eindimensionale Eigenschaft von Systemen, sondern hat verschiedene, ihr zugrunde liegende Komponenten. Die Unterteilung in Kriterien der Usability (einen Überblick gibt Richter, 1997) wird unterschiedlich vorgenommen – hier zwei Beispiele:

- Nielsen (1993, S. 26) gibt die folgenden *fünf* Usability-Kriterien an: Learnability, Efficiency, Memorability, Errors and Satisfaction.
- In der ISO-Norm 9241-10 «Ergonomische Anforderungen für Bürotätigkeiten mit Bildschirmgeräten – Teil 10: Grundsätze der Dialoggestaltung» (International Organization for Standardisation, 1996) werden die folgenden *sieben* Kriterien unterschieden: Aufgabenangemessenheit, Selbstbeschreibungsfähigkeit, Steuerbarkeit, Erwartungskonformität, Fehlertoleranz, Individualisierbarkeit und Lernförderlichkeit.

2.3. User Interface

Hier soll zuerst der Begriff *User Interface* definiert und verschiedene *User Interface Paradigmen* beziehungsweise *Interaktionsstile* vorgestellt werden.

2.3.1. Begriffsdefinition

Das User Interface ist jener Teil eines technischen Systems, über welchen ein Benutzer mit dem System interagiert. Es kann davon ausgegangen werden, dass ein Endbenutzer das User Interface mit dem gesamten System gleichsetzt. Diese Schnittstelle betrifft gleichwohl Hardware und Software, wobei der Begriff bei Computersystemen häufig nur für den Software-Teil des Systems verwendet wird. Obwohl auch der Software-Teil des User Interface verschiedenste Aspekte hat, wird häufig vor allem auf den visuellen Aspekt der Informationspräsentation Bezug genommen. Genauere Definitionen sind zu finden z.B. in Mandel (1997, S. 9) oder Mayhew (1992, S. 7).

Heute sind vor allem die Graphical User Interfaces verbreitet, das war aber nicht schon immer so. Früher herrschten andere User Interface Paradigmen vor.

2.3.2. Verschiedene User Interface Paradigmen – Interaktionsstile

Verschiedenen User Interfaces liegen grundsätzliche Interaktionsstile zu Grunde, die auch als User Interface Paradigmen bezeichnet werden können. Ein User Interface beinhaltet in den seltensten Fällen nur einen einzigen Interaktionsstil. Meist ist aber ein Stil in einem Interface vorherrschend. Unten werden die Paradigmen *Kommandozeile*, *Menübasiertes Interface*, *Direkt Manipulation / GUI* sowie *Web Interface* genauer beschrieben.

Parallel zu diesen Interaktionsstilen kann noch das Ausführen von Kurzbefehlen über das Betätigen von Tastenkombinationen oder Funktionstasten genannt werden, welches dem geübten Benutzer erlaubt, einzelne Funktionen effizient über die Tastatur aufzurufen. Ein Nachteil dieser Methode ist, dass der Benutzer die jeweiligen Tasten auswendig wissen muss. Da die Tastenkombinationen aber selten als einzige Möglichkeit der Interaktion eingesetzt werden, ist dieses Problem nicht so gravierend. Ein Beispiel aus Mac OS ist das gleichzeitige Drücken der «Befehlstaste» und der Taste «s», um das aktive Dokument zu sichern:



2.3.2.1. Kommandozeile

Frühere Computer wurden üblicherweise über eine Kommandosprache bedient, die spezifisch für das jeweilige Betriebssystem definiert war. Der Computer fordert den Benutzer mit dem sog. «Ready Prompt» auf, die entsprechenden Befehle einzugeben. So werden z.B. die Benutzer von DOS «begrüsst» mit einem «C:\>_». Der Benutzer gibt einen (oder mehrere, aneinander gehängte) Befehle ein und schliesst die Eingabe mit der Enter-Taste ab. Der Computer führt den Befehl aus, zeigt das Ergebnis an und fordert den Benutzer mit einem weiteren Ready Prompt wieder zu Eingaben auf. In Abbildung 4 ist eine solche Abfolge zu sehen. Einige weitere DOS Befehle sind: APPEND, CHKDSK, COPY und MKDIR.

```

Microsoft(R) Windows 98
(C)Copyright Microsoft Corp 1981-1999.

C:\>dir temp

Datenträger in Laufwerk C: hat keine Bezeichnung
Seriennummer des Datenträgers: 0000-6BEC
Verzeichnis von C:\Temp

.                <DIR>           05-28-02  1:33p  .
..               <DIR>           05-28-02  1:33p  ..
INDEX~1  HTM           0  05-28-02  1:33p  index.html
TEST     TXT           0  05-28-02  1:33p  Test.txt
ABCDE    TXT           0  05-28-02  1:35p  abcde.txt
          3 Datei(en)                0 Bytes
          2 Verzeichnis(se)  1,723,531,264 Bytes frei

C:\>

```

Abbildung 4 DOS Kommandozeile (Screenshot aus Windows 98)

Für Anfänger und Gelegenheitsbenutzer ist diese Art der Interaktion sehr mühsam. Wer die Befehle nicht auswendig weiss, muss jeden einzelnen in der Dokumentation nachschlagen. Für geübte Benutzer kann die Bedienung über eine Kommandozeile jedoch sehr effizient sein. Mayhew (1992, S. 228–265) fasst Hinweise für die Gestaltung von Kommandosprachen zusammen.

2.3.2.2. Menübasiertes Interface

In einem User Interface, welches *auf Menüauswahl basiert*, ist die hauptsächliche Form von Interaktion eine Sequenz, in welcher dem Benutzer jeweils eine Auswahl präsentiert wird. Über die Auswahl einzelner Optionen kann so durch die verschiedenen Bereiche navigiert werden. In Abbildung 5 ist ein Beispiel einer menübasierten Applikation zu sehen.

Mayhew (1992) nennt verschiedene Vorteile, die menübasierte Systeme haben können:

- *Selbsterklärend*: Mit gut gestalteten Menüs sind Systeme sehr einfach zu lernen, da die Menüs explizit machen, was mit einem System gemacht werden kann (Semantik) und wie es getan werden kann (Syntax).
- *Beansprucht das Gedächtnis wenig*: Da das Menü die Optionen anzeigt, wird der Benutzer durch das System an die Möglichkeiten erinnert (remind) anstatt, dass er sich selber an mögliche Befehle erinnern muss (remember).
- *Weniger Tasteneingaben*: Der Benutzer kann über ganz wenige Eingaben an der Tastatur oder z.B. über Anklicken mit einer Maus eine Auswahl treffen, was weniger Zeit braucht und auch weniger Möglichkeiten für Fehler bietet.
- *Einfache Fehlerbehandlung*: Durch die limitierten Möglichkeiten einer Eingabe ist eine gute Fehlerbehandlung relativ einfach.
- *Erweiterungen sind sichtbar*: Da das Menü Interface immer alle möglichen Optionen präsentiert, ist sofort sichtbar, wenn das System um neue Funktionen erweitert wurde.

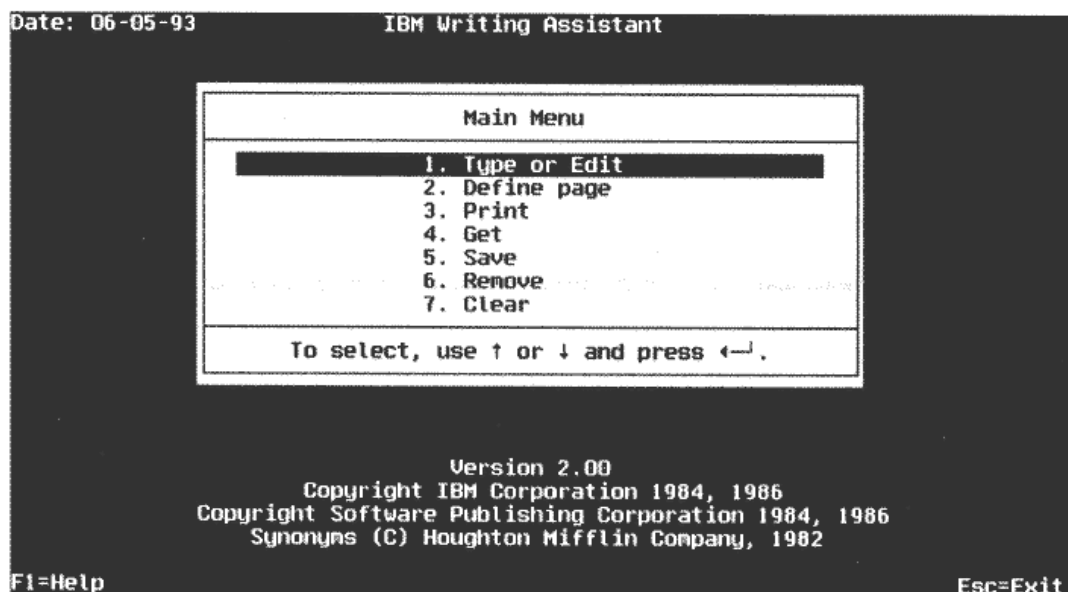


Abbildung 5 Menübasierte Anwendung von IBM (aus Mandel, 1997)

Als Nachteile menübasierter Systeme führt Mayhew (1992) an:

- *Ineffizient*: Müssen Menüs durch die grosse Anzahl von Funktionen hierarchisch aufgebaut werden, kann das sehr schnell ineffizient werden.

- *Unflexibel*: Das System gibt die Struktur vor und der Benutzer muss sich an den vorgegebenen Aufbau halten. Dies kann sehr hinderlich sein, es sei denn, die Aufgabe selbst ist schon sehr strukturiert.
- *Unpraktisch für sehr viele Optionen*: Was bei einer überschaubaren Zahl von Optionen noch gut funktioniert, wird bei sehr vielen Optionen undurchschaubar oder sogar unmöglich.
- *Beansprucht Platz auf dem Bildschirm*: Vor allem wenn sie viele Optionen enthalten, benötigen Menüs Platz auf dem Bildschirm, der dem Benutzer nicht mehr für andere Dinge zur Verfügung steht.

Menübasierte User Interfaces werden nebst Computern vor allem auch bei anderen Geräten wie Mobiltelefonen, Geldautomaten, telefonischen Reservationssystemen etc. eingesetzt. Ein Beispiel von einem Mobiltelefon ist in Abbildung 6 zu sehen.



Abbildung 6 Menü eines Mobiltelefons (von www.nokia.com)

2.3.2.3. Direktmanipulation / Graphical User Interface (GUI)

Direktmanipulation ist *eines* der grundlegenden Prinzipien des Graphical User Interface (GUI)² von aktuellen Betriebssystemen wie dem Mac OS von Apple Computer, Inc. Die Verwendung der sog. «Desktop Metaphor» ist ein weiteres Prinzip. Weitere sind als «Human Interface Design Principles» bei Apple Computer, Inc. (1995) aufgeführt.

² Der Einfachheit halber wird in dieser Arbeit der Begriff «GUI» mit Systemen gleichgesetzt, die über ein Graphical User Interface mit «Desktop Methaphor» verfügen. Solche Interfaces werden auch WIMP Interfaces (Windows, Icons, Menus and Pointing devices) genannt.

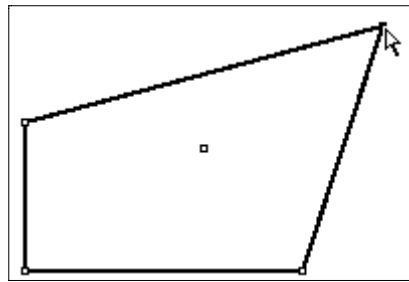


Abbildung 7 Beispiel von Direktmanipulation (Screenshot aus Illustrator)

Der Benutzer eines GUI kann nach dem Prinzip der Direktmanipulation viele Objekte direkt bearbeiten. Das Beispiel in Abbildung 7 zeigt das Verändern eines Polygons mit dem Mauszeiger. Es können auch Dateien mittels «Drag and Drop» innerhalb des Filesystems verschoben werden etc. Das Prinzip der Direktmanipulation wird von Hutchins, Hollan und Norman (1986) genauer besprochen.

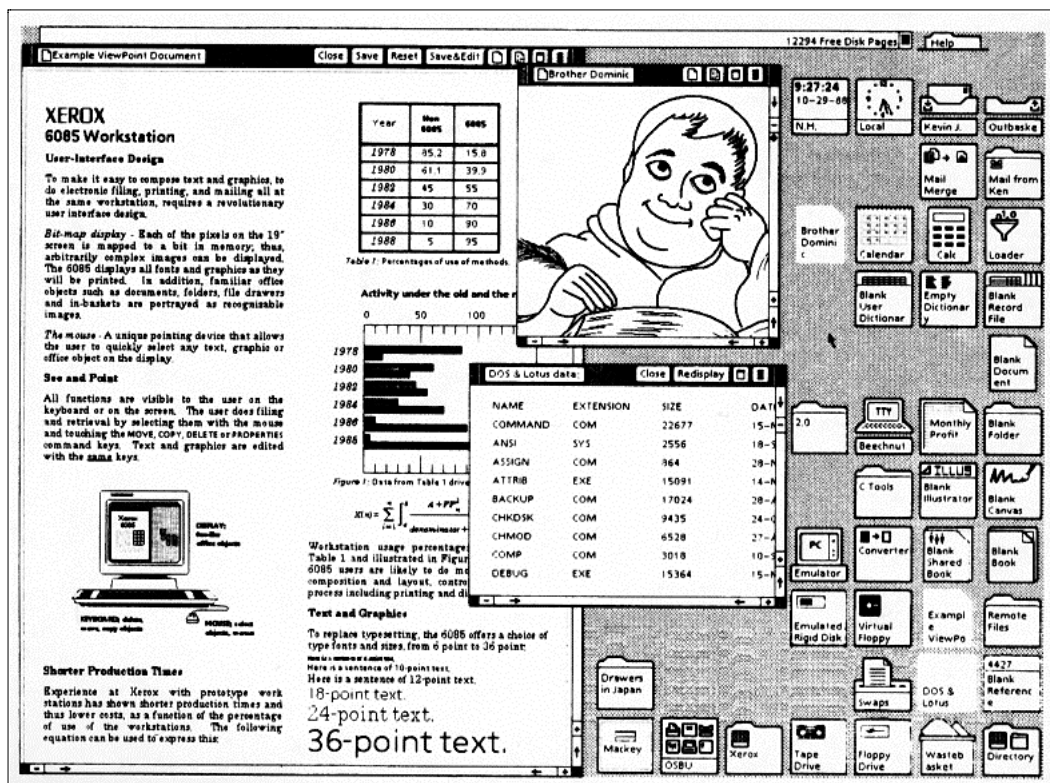


Abbildung 8 GUI von Xerox Star (aus Johnson et al., 1995)

Die Grundlagen von GUIs gehen zurück auf Arbeiten aus den Jahren um 1960 von Ivan Sutherland am MIT und Douglas Engelbart in Stanford. 1981 wurde bei Xerox der erste kommerzielle Computer mit einem GUI produziert: Xerox Star (Johnson, Roberts, Verplank, Smith, Irby, Beard und Mackey, 1995). Das GUI des Star ist in Abbildung 8 zu sehen.

Trotz Direktmanipulation werden in GUIs verschiedenste Arten von Menüs eingesetzt, durch die grafischen Möglichkeiten der Systeme können diese jedoch viel flexibler gestaltet und wiederum über Direktmanipulation bzw. nach dem Prinzip «See-and-point» bedient werden.

Eine Art von Menü – die Pull-down Menüs³ – befindet sich in der Menüleiste Ein Beispiel ist in Abbildung 9 zu sehen.

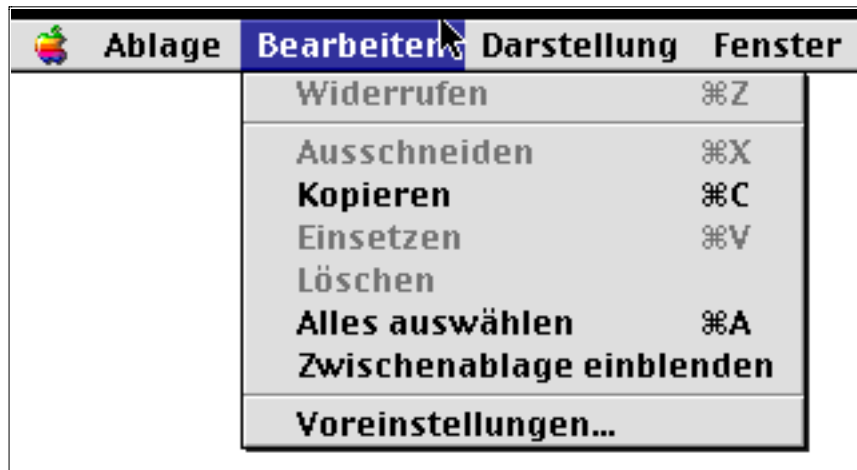


Abbildung 9 Menü «Bearbeiten» in der Menüleiste (Screenshot aus Mac OS)

Über die Pull-down Menüs können weiter z.B. Dialogboxen aufgerufen werden, in welchen über Pop-up Menüs weitere Einstellungen getroffen werden. Ein Beispiel dazu ist in Abbildung 10 zu sehen.

³ Plattformspezifische Begriffe: Zur Vereinfachung und somit besseren Lesbarkeit des Textes werden in dieser Arbeit Begriffe des Mac OS verwendet. So ist z.B. mit «Pull-down Menü» ein Menü in der Menüleiste gemeint, unabhängig von der Plattform.

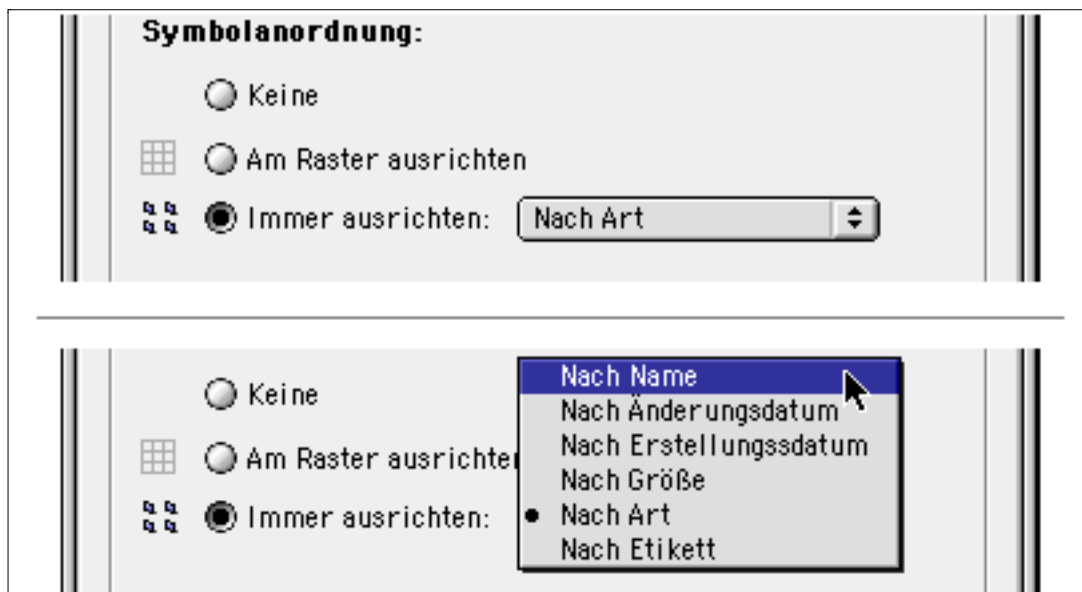


Abbildung 10 Pop-up Menü in einer Dialogbox (Screenshot aus Mac OS)

Dem Benutzer können auch gleich bei einzelnen GUI-Elementen Kontextmenüs zur Verfügung gestellt werden. Diese Menüs beinhalten nur jene Optionen, die auf das entsprechende Objekt angewendet werden können, wie das auch am Beispiel in Abbildung 11 zu sehen ist.

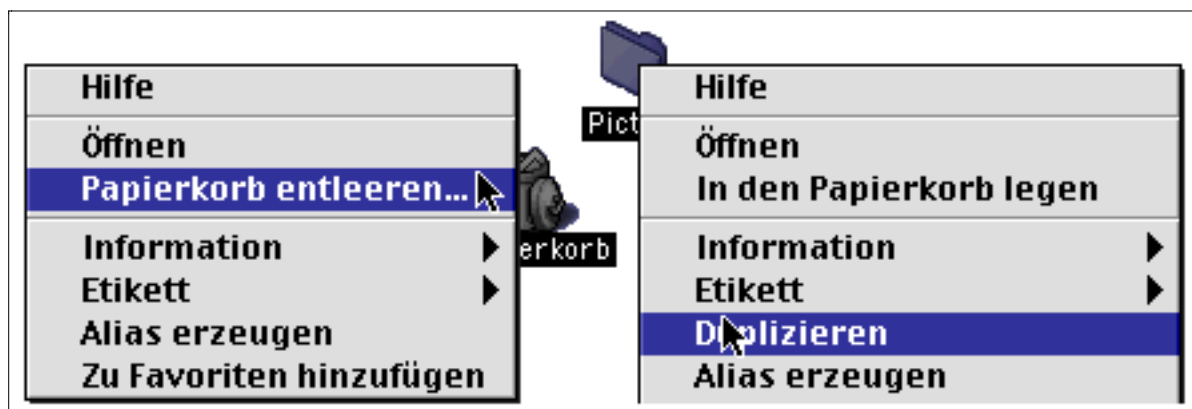


Abbildung 11 Kontextmenüs (Screenshot aus Mac OS)

Dank der grafischen Möglichkeiten von GUIs sind die verwendeten Menüs auch nicht auf die Gestalt von Listen beschränkt. Die ganze Vielfalt aller verschiedenen Arten von Menüs kann hier nicht besprochen werden, da auch Elemente wie Toolbars etc. als eine Variante von Menüs bezeichnet werden können. So werden GUIs über die Zeit auch immer weiter mit neuen Arten von Menüs ergänzt. Eine solche Entwicklung kann bei Apple Computer, Inc. (1987, 1995, 1997 und 2001) nachvollzogen werden.

Apple Computer, Inc. (1987, 1995, 1997 und 2001) gibt – vor allem in Bezug auf die Menüleiste – genau vor, wie die Menüs zu gestalten sind und diese Vorgaben werden von anderen

Anbietern von Macintosh Software ziemlich gut befolgt. Für andere Plattformen existieren ähnliche Vorgaben (z.B. Sun Microsystems, Inc., 2001 oder Microsoft Corp., 1999). Weitere Hinweise zur Gestaltung von Menüs in GUIs sind bei Cooper (1995) oder Johnson (2000) zu finden.

2.3.2.4. Web Interface

Im Internet in Form des World Wide Web aber auch in unternehmensinternen Intranets nimmt die Bedeutung von webbasierten User Interfaces ständig zu.

Je nach verwendeter Technologie sind Webseiten mehr oder weniger interaktiv. So gleicht das Verhalten von Webseiten, die nur mit HTML realisiert sind, eher den Menübasierten Interfaces (siehe Abschnitt 2.3.2.2, Seite 9), da die Navigation lediglich über Hyperlinks «von Seite zu Seite» geschieht. Werden nebst HTML weitere Technologien wie JavaScript, Flash oder Java eingesetzt, kann mit den Webseiten viel flexibler interagiert werden. Der Einsatz von Menüs kann dabei zunehmend dem bei GUIs (siehe Abschnitt 2.3.2.3, Seite 11) gleichen.



Abbildung 12 Beispiele von Menüs auf www.cnn.com und www.yahoo.com

Die Gestaltung von Menüs auf Webseiten ist in keiner Art geregelt, vielmehr kann sie sehr unterschiedlich aussehen, wie auch in Abbildung 12 und Abbildung 13 zu sehen ist.

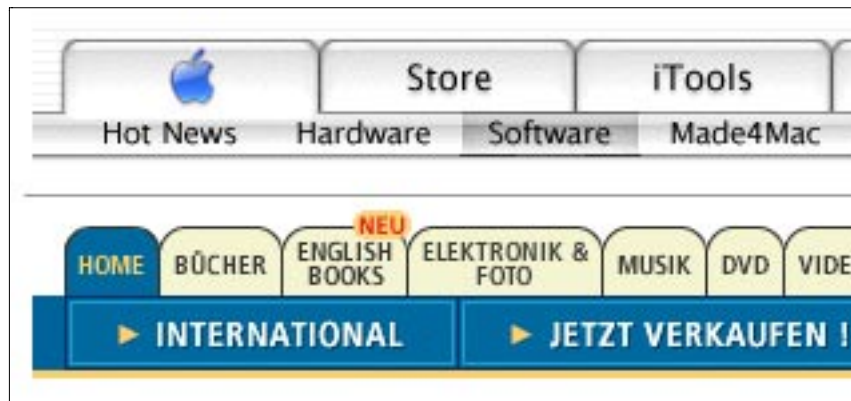


Abbildung 13 Beispiele von Menüs auf www.apple.com und www.amazon.de

Es gibt aber diverse Autoren, die zu Menüs in Web Interfaces Empfehlungen abgeben (siehe dazu z.B. Lynch und Horton, 1999; Fleming, 1998; Nielsen, 1999; Spool, Scanlon, Schroeder, Snyder und De Angelo, 1999). Weiter findet in letzter Zeit das Gebiet «Information Architecture» Verbreitung, welches sich im Bereich Web gemäss Rosenfeld und Morville (1998) auf Organisation, Indexierung und Navigationssysteme spezialisiert hat.

3. Klassische Fragen der Menüforschung

Kent Norman (1991) hat in seinem Buch «The psychology of menu selection» versucht, den damaligen Wissensstand zum Thema Menüs möglichst umfassend zusammenzutragen. Weitere Übersichten zur Forschung in Bezug auf Menüs gibt es von Mayhew (1992, Kap. 4, S. 113–171) sowie Paap und Cooke (1997).

Nebst anderem waren vor allem die *Strukturierung* und die *Gestaltung von Menüs* Themen vieler Forschungsbemühungen.

3.1. Strukturierung von Menüs

In Bezug auf Strukturierung von Menüs werden hier die *Struktur über mehrere Menüs hinweg* wie auch die *Struktur innerhalb eines einzelnen Menüs* besprochen.

3.1.1. Struktur über mehrere Menüs

Ein Thema, das zu vielen Untersuchungen geführt hat, ist der sogenannte «depth-breadth trade-off», das heisst die Frage, ob es besser ist, weniger Optionen in einzelnen Menüs aber dafür tiefere Hierarchien oder flachere Hierarchien mit je mehr einzelnen Optionen zu haben. Eine ausführliche Besprechung der Forschung zu diesem und zu anderen Themen in Bezug auf die Strukturierung von Menüs ist zu finden bei Norman (1991), Mayhew (1992) sowie Paap und Cooke (1997).

Mayhew (1992, S. 144–154) hat aus den Forschungsergebnissen folgende *Hinweise zur Strukturierung von Menüs* zusammengestellt:

- Match menu structure to task structure.
- Provide users with an easy way to tailor menu structure to task structure.
- Minimize menu hierarchy depth at the expense of breath.
- On full-screen text menus, present menu choice lists vertically.
- Consider pie menus for one- or two-level mouse driven menu hierarchies with short menu choice lists, especially when the choices lend themselves well to a circular, pie format.
- Choosing graying out or deletion of inactive menu items depends on user experience and input device.
- Create logical, distinctive, and mutually exclusive semantic categories with clear meanings.
- Menu choice labels should be brief, consistent in grammatical style and placement, and matched with corresponding menu titles.
- Consider menu choice descriptors in choice labels may not be clear and unambiguous.

Weiter sind einige Regeln zur Strukturierung von Menüs in GUIs in den entsprechenden Guidelines (z.B. Apple Computer, Inc., 1995) teilweise festgeschrieben und einiges spricht

dafür, diesen Vorgaben zu folgen, damit eine gewisse Konsistenz gewährleistet werden kann. Trotzdem sind diese Regeln auch nicht ganz unbestritten (siehe dazu Cooper, 1995).

3.1.2. Struktur innerhalb eines Menüs

Zur Organisation der Menüoptionen, z.B. die Reihenfolge der Einträge eines einzelnen Menüs, wurde ebenfalls viel Forschung betrieben – auch hierzu geben Norman (1991), Mayhew (1992) sowie Paap und Cooke (1997) einen Überblick.

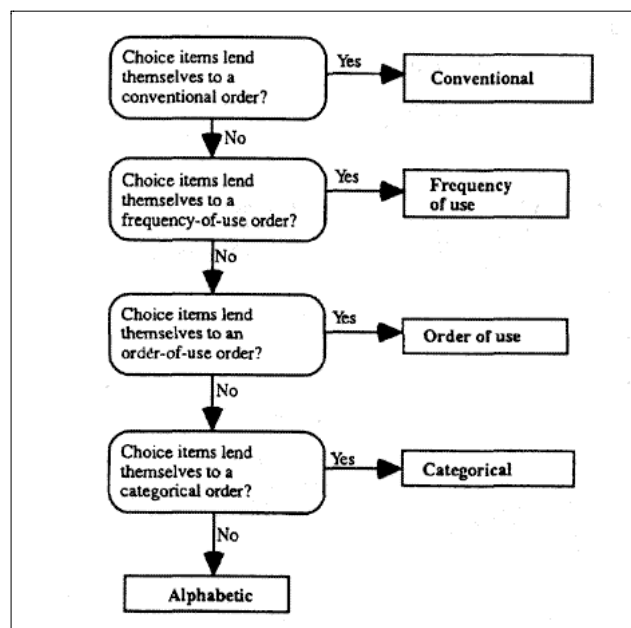


Abbildung 14 Ordnungsschemata für Menüs (aus Mayhew, 1992)

Mayhew (1992, S. 154) kommt in Bezug auf Ordnung der Menüoptionen zu folgenden Regeln: «Order menu choice labels according to convention, frequency of use, order of use, categorical or functional groups, and/or alphabetical order, depending on the user and task variables.» In Abbildung 14 ist diese Regel als Diagramm zu sehen.

3.2. Präsentation von Menüs

Auch zur Präsentation von Menüs wurden verschiedene Untersuchungen durchgeführt. Zur Gestaltung von Menüs in GUIs existieren vor allem in Bezug auf das Fitts's Law verschiedene Publikationen. Hier soll zuerst das wichtige Gesetz *Fitts's Law* vorgestellt und nachher einige entsprechende *Optimierungen von Menüs* beschrieben werden.

3.2.1. Fitts's Law

Ein wichtiges Gesetz für Fragen der Gestaltung von User Interfaces wurde von P. M. Fitts formuliert (siehe z.B. Raskin, 2000 oder Card, Moran und Newell, 1983):

Je weiter ein Ziel von der ursprünglichen Cursorposition weg oder je kleiner das Ziel ist, desto länger wird es dauern, den Cursor zum Ziel zu bewegen.

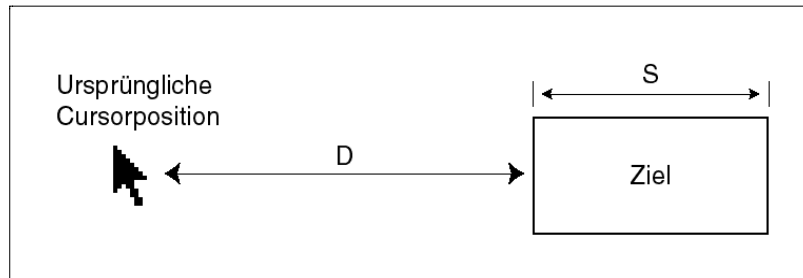


Abbildung 15 Fitts's Law schematisch (nach Raskin, 2000)

In Abbildung 15 ist die Situation schematisch aufgezeichnet. Die Länge einer geraden Linie von der ursprünglichen Cursorposition bis zum nächstliegenden Punkt auf dem Ziel definiert die *Distanz* (D). Die *Grösse* (S) des Ziels wird entlang der Bewegungslinie gemessen.

Nach Fitts's Law (nach Raskin, 2000) ergibt sich die durchschnittliche Zeit für die erfolgreiche Bewegung:

$$\text{Zeit (in msec)} = a + b \log_2 (D / S + 1)$$

(Die Konstanten a und b sind experimentell zu bestimmen oder aus bekannten Parametern abzuleiten).

Soviel zur Theorie – unten soll mit einigen Beispielen illustriert werden, wie diese Erkenntnis auf das Design eines User Interface angewendet werden kann.⁴

3.2.2. Optimierung von Menüs

Es bestehen unzählige Präsentationsformen für Menüs. Unten sollen einige Beispiele zeigen, wie – unter Berücksichtigung des Fitts's Law – die Interaktion mit den Menüs optimiert werden kann. Das *Plazieren eines Menüs am Bildschirmrand* erreicht eine virtuelle Vergrößerung des Ziels. Auch durch ein *Optimieren von hierarchischen Menüs* lassen sich Menüs verbessern – weitere solche Beispiele sind bei Tognazzini (1999) zu finden.

⁴ Leser, die bereits hier denken, dass ihnen alles zu diesem Gesetz klar ist, können sich vor dem Weiterlesen am Quiz von Tognazzini (1999) versuchen.

Menü am Bildschirmrand platzieren

Durch die Platzierung am Bildschirmrand bekommt ein Menü eine nicht passierbare Abgrenzung und wird dadurch virtuell grösser. Walker und Smelcer (1990) sowie Tognazzini (1993) haben diese Faktoren untersucht.



Abbildung 16 Menüleiste von Mac OS (Screenshot aus Mac OS)

In Abbildung 16 ist die Menüleiste des Mac OS und in Abbildung 17 die einer Windows Applikation zu sehen. Wie in den Abbildungen zu sehen ist, hat bei Windows das Ziel nur eine Höhe von 18 Pixel, was eine langsame, kontrollierte Bewegung erfordert. Bei Mac OS kommt zu den gleichen 18 Pixel Höhe noch eine unendlich grosse Verlängerung hinzu, da der Cursor nach Erreichen des Bildschirmrands im Menü stehen bleibt. Der Benutzer kann ein Menü durch eine sehr schnelle, ballistische Bewegung ansteuern.

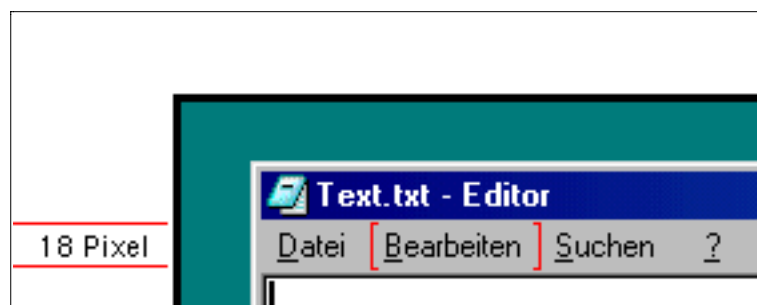


Abbildung 17 Menüleiste aus Windows'98 Applikation (Screenshot aus Windows'98)

Nach Berechnungen von Raskin (2000) ist der Benutzer eines Macintosh ca. 2,5 mal schneller beim Erreichen des Menüs als der Benutzer eines Windows PC.

Optimieren von hierarchischen Menüs

Abbildung 18 zeigt das Beispiel eines hierarchischen Menüs. Bei vielen Systemen sind solche Menüs so implementiert, dass der Benutzer mit dem Mauszeiger eine L-förmige Bewegung machen muss, um einen Menüpunkt des Untermenüs wählen zu können, d.h. er müsste den Cursor auf dem Menüpunkt «Anordnen» zuerst genau horizontal nach rechts bewegen, bis das Untermenü («Nach Name, Nach ...») erreicht ist, sonst würde es zuklappen.

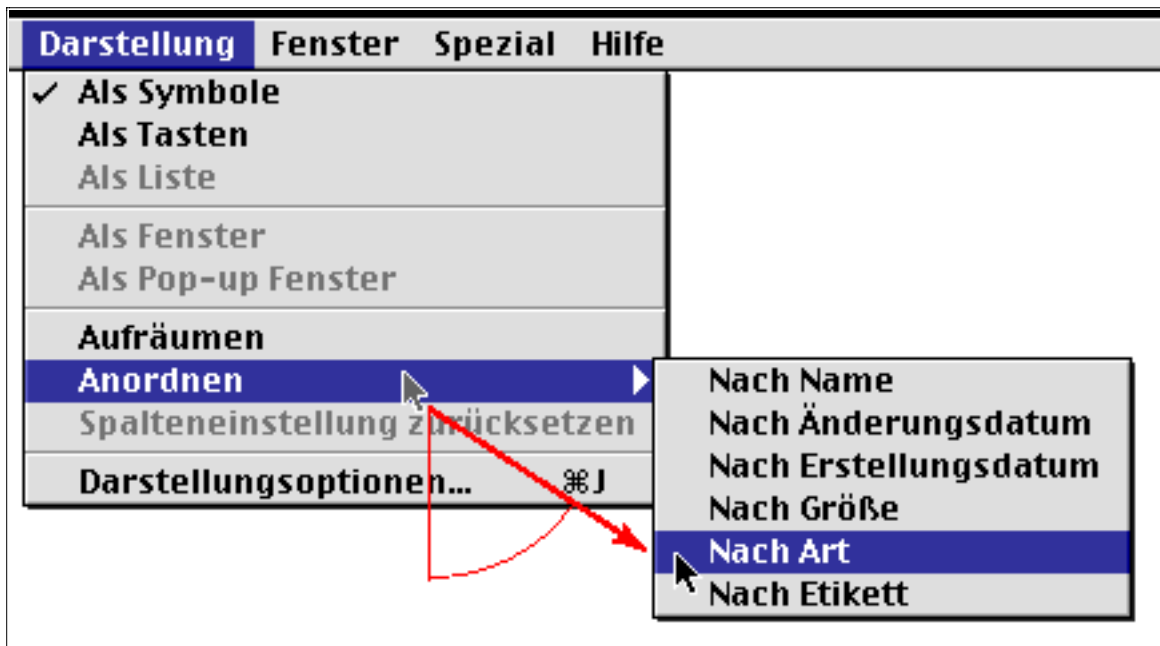


Abbildung 18 Hierarchisches Pull-down Menü (Screenshot aus Mac OS)

Da dies jedoch motorisch sehr anspruchsvoll ist, sind die hierarchischen Menüs z.B. in Mac OS so implementiert, dass bei einer Bewegung mit einem gewissen Winkel nach unten und rechts – wie in Abbildung 18 zu sehen – das Untermenü nicht zuklappt.

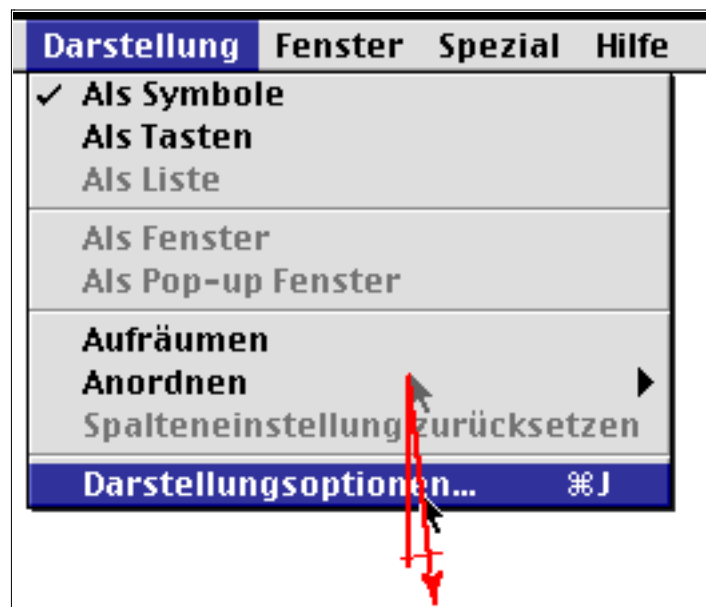


Abbildung 19 Auswahl eines Menüpunktes (Screenshot aus Mac OS)

Will der Benutzer jedoch eine Menüpunkt weiter unten anwählen (Beispiel in Abbildung 19: «Darstellungsoptionen ...») und ist der Winkel dadurch sehr klein, schliesst sich das Untermenü sofort wieder (siehe dazu Tognazzini, 1999).

Eine andere Lösung für das Problem der Navigation in hierarchischen Menüs, die z.B. in Windows 95 angewandt wurde, ist jene, die Untermenüs in jedem Fall erst nach einer Zeitverzögerung wieder zu schliessen. Dies führt jedoch dazu, dass jedes Untermenü für eine kurze Zeit geöffnet bleibt, auch wenn ein Benutzer mit dem Mauszeiger nur gerade nach unten fährt. In Web Interfaces mit dieser Art Untermenüs wird dem Problem oft gar keine Beachtung geschenkt.

Pie Menü

Eine weitere Möglichkeit, die Erkenntnisse aus dem Fitts's Law in einem Menü umzusetzen, ist der Einsatz sogenannter Pie Menüs. Statt in Listenform werden die Menüpunkte kreisförmig angeordnet. Ein einfaches Beispiel ist in Abbildung 20 zu sehen. Pie Menüs können aber auch hierarchisch aufgebaut sein.

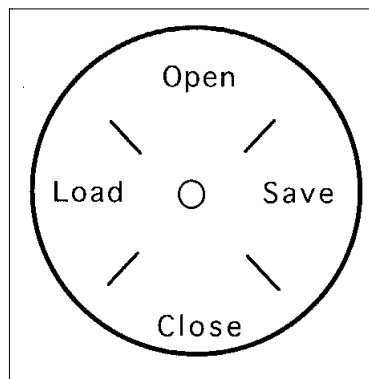


Abbildung 20 Pie Menü (aus Shneiderman, 1998)

Da sich der Cursor beim Öffnen des Menüs in der Mitte befindet, ist die Distanz zu allen einzelnen Menüpunkten minimal. Auch bieten die Kreissegmente ein grösseres Ziel. Untersuchungen haben gezeigt, dass Pie Menüs im Vergleich zu linearen Menüs einen Geschwindigkeitsvorteil bringen können (Callahan, Hopkins, Weiser und Shneiderman, 1988; Mills und Prime, 1990). Allerdings muss auch bedacht werden, dass die kreisförmige Anordnung auch Nachteile bringen können: Zum Beispiel finden anzahlmässig weniger Einträge Platz und die Menüeinträge oben beziehungsweise unten nehmen auf dem Kreis mehr Platz weg.

4. Neuere Menüforschung

Hier sollen als erstes die Themen der aktuellen Menüforschung präsentiert werden. Am Beispiel der 3-D Animationssoftware Maya soll danach aufgezeigt werden, dass auch ungewöhnlichere Menükonzepte erfolgreich in kommerzieller Software eingesetzt werden.

4.1. Aktuelle Themen der Forschung

Eine Frage, die nach längerer Zeit wieder aufgegriffen wurde, ist die nach der *Strategie*, welche Benutzer anwenden, wenn sie einen bestimmten *Menüeintrag suchen*. Zu diesem Thema sollen hier einige Untersuchungen mit den Methoden kognitive Modellierung und Eyetracking vorgestellt werden.

Die heutigen Computeranwendungen stellen immer wieder neue Anforderungen an Menüs, und die HCI-Forschung versucht, geeignete Lösungen zu finden. Die verschiedenen Themenbereiche werden unten vorgestellt. Ein wichtiger Aspekt dabei ist, Designs für eine *große Anzahl von Menüoptionen* zu finden. Der Versuch, Menüs möglichst im Kontext der Arbeit anbieten zu können, führte zu Forschungen mit *transparenten Menüs*. Mit den *Marking Menüs* wurde eine Weiterentwicklung zu den schon länger bekannten Pie Menüs vorgestellt.

4.1.1. Suche innerhalb von Menüs

Eine Reihe von Untersuchungen hat sich in jüngster Zeit der Frage angenommen, wie Benutzer nach einem Menüeintrag in einem Pull-down Menü suchen.

Mit Hilfe von kognitiver Modellierung sind Hornof und Kieras (1997) zum Schluss gekommen, dass Benutzer mehrere Menüeinträge gleichzeitig verarbeiten und Menüs sowohl mit zufälligen als auch mit systematischen Strategien absuchen. Hornof und Kieras (1999) entwickelten auch ein Modell, um die Suche nach bereits bekannten Menüoptionen erklären zu können.

Aaltonen, Hyrskykari und Rähä (1998) haben mittels Eyetracking zeigen können, dass Benutzer Menüs mit sequentiellen Gleitbewegungen (sweeps) absuchen, was eine Erklärung für die von Hornof und Kieras (1997) gefundenen hybriden Strategien sein könnte. Byrne, Anderson, Douglass und Matessa (1999) sehen nach ihren Eyetracking-Versuchen jedoch in allen bisher vorgeschlagenen Modellen noch keine Antwort auf die Frage nach dem Suchverhalten und fordern, dass die Frage weiter untersucht werden sollte.

4.1.2. Menüs für eine grosse Menge von Optionen

Eine der grossen Herausforderungen ist die Entwicklung von Menüs für eine grosse Anzahl von Optionen. *Split Menüs* und *Fisheye Menüs* sind Ansätze, die als herkömmliche lineare Menüs umgesetzt wurden. Aus dem Gebiet der *Information Visualization* gibt es unzählige neue Techniken, um in grossen Mengen von Daten zu navigieren und Auswahlen zu treffen.

4.1.2.1. Split Menüs

Sollen in einem umfangreichen Menü einzelne Menüpunkte immer wieder ausgewählt werden, andere aber eher selten, könnte ein sogenanntes Split Menü eingesetzt werden. Wie in Abbildung 21 am Beispiel eines Schrift-Menüs zu sehen ist, wird das Menü in zwei Teile aufgeteilt. Im ersten Teil des Menüs befinden sich die oft verwendeten Menüoptionen, die anderen befinden sich anderen Bereich weiter unten.

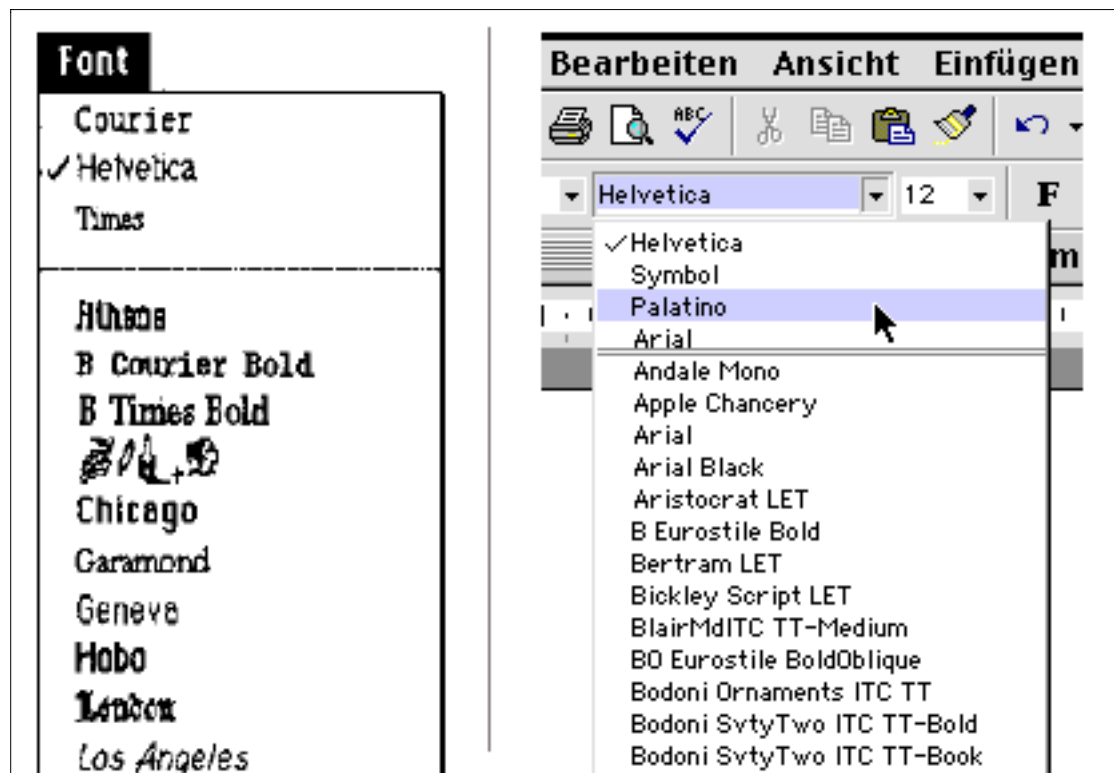


Abbildung 21 Split Menüs (aus Sears und Shneiderman, 1994 und Screenshot aus Word)

Sears und Shneiderman (1994) konnten in Felduntersuchungen zu Split Menüs Geschwindigkeitsvorteile von 17 bis 58% beobachten. In einem kontrollierten Experiment waren die Split Menüs signifikant schneller und wurden auch in einer subjektiven Bewertung den rein alphabetischen Menüs vorgezogen.

4.1.2.2. Fisheye Menüs

Eine Möglichkeit, detaillierte Informationen in ihrem Kontext anzuzeigen, ist die sogenannte Fisheye Ansicht. Furnas (1986) führte dieses Konzept ein, welches bereits auf Tabellen, Diagramme, Karten etc. angewandt wurde.

Bederson (2000) stellte das *Fisheye Menü* vor – ein Beispiel ist in Abbildung 22 (auf Seite 25) zu sehen. Eine erste Evaluation hat gezeigt, dass Fisheye Menüs in bestimmten Situationen (bei sogenannten «browsing tasks») anderen Menüarten vorgezogen wurden.

4.1.2.3. Ansätze aus dem Gebiet Information Visualization

Information Visualization ist ein grosses HCI Forschungsgebiet, welches sich mit der Visualisierung grosser Datenmengen befasst und sich auch schon wieder in Teilgebiete aufzuteilen beginnt. Einen Überblick zur aktuellen Forschung des ganzen Gebiets geben Card, Mackinlay und Shneiderman (1999). Einen Überblick zu einem der Teilgebiete, der Repräsentation und Navigation von Graphen – d.h. strukturierter Daten – geben Herman, Melançon und Marshall (2000).

Das Gebiet wird immer grösser und die Abgrenzung zum Thema der Menüs ist verschwommen. Hier soll anhand einiger Beispiele exemplarisch aufgezeigt werden, welcher Art die Lösungen in Bezug auf Navigation und Auswahl von Optionen sind.

Cone Tree

Robertson, Mackinlay und Card (1991) haben mit den Cone Trees eine Technik vorgestellt, Hierarchien dreidimensional platzsparend zu visualisieren.

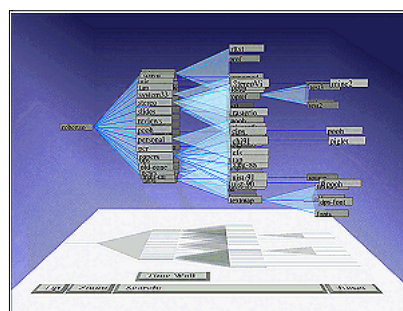


Abbildung 23 Cone Tree (von www.inxight.com)

Ein Beispiel einer solchen Ansicht ist in Abbildung 23 zu sehen. Der Einsatz interaktiver Animation bei den Cone Trees hilft, einen Teil der kognitiven Belastung auf das Wahrnehmungssystem zu verlagern.

Table Lens

Mit Table Lens präsentierten Rao und Card (1994) eine neue Technik zur Visualisierung grosser Tabellen. Mit der verwendeten Fisheye Technik können verschiedene fokale Bereiche bestimmt werden – im Beispiel in Abbildung 24 sind in zwei Bereichen die Daten von je fünf Zeilen zu sehen.

	Avg	Career Avg	Team	Salary 87
Larry Herndon	0.24734983	0.27282876	Det	225
Jesse Barfield	0.2886248	0.27269818	Tor.	1137.5
Jeffrey Leonard	0.27859238	0.27260458	S F	900
Donnie Hill	0.28318584	0.2725564	Dak.	275
Billy Sample	0.285	0.2718601	Atl.	MA
Howard Johnson	0.24514455	0.25232068	N Y	257.5
Andre Thomas	0.250774	0.2521994	Atl.	75
Billy Hatcher	0.25775656	0.25211507	Hou.	110
Omar Moreno	0.2339833	0.2518029	Atl.	88
Darnell Coles	0.2725528	0.25153375	Det.	105

Abbildung 24 Table Lens (aus Rao und Card, 1994)

Die Ansicht vereinigt symbolische und grafische Repräsentationsformen, die durch den Benutzer angepasst werden können. Die Daten können durch Direktmanipulation einfach exploriert und analysiert werden.

Treemaps

Mit Tree-Maps haben Johnson und Shneiderman (1991) eine Form der Visualisierung präsentiert, die Hierarchien auf ein rechteckiges Fenster abbilden.

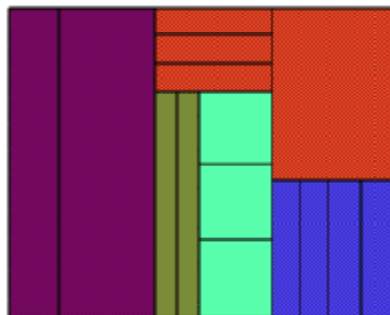


Abbildung 25 Treemap schematisch (aus Herman et al., 2000)

Abbildung 27 zeigt ein Menü mit Click-trough Buttons, die es erlauben, das darunterliegende Objekt mit einer bestimmten Füllfarbe bzw. einer Konturfarbe zu versehen. Die Bedienung geschieht dabei mit zwei Händen, wobei mit der einen Hand (z.B. mit einem Trackball) der gewünschte Button über dem Objekt positioniert und mit der anderen Hand (z.B. mit der Maus) die Auswahl getätigt werden kann.

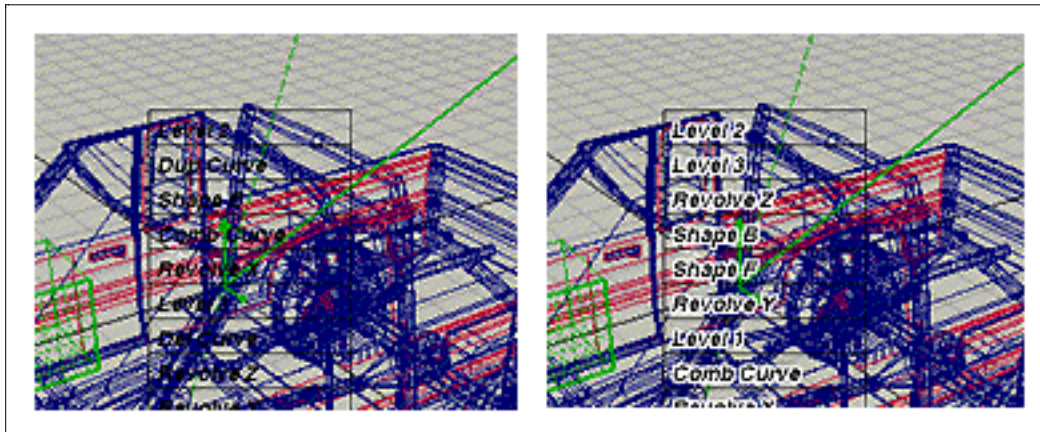


Abbildung 28 Transparente Menüs mit verschiedenen Schriften (aus Harrison und Vicente, 1996)

Auch bei eher konventionellen Kontextmenüs kann Transparenz dazu eingesetzt werden, dass der Benutzer den Kontext seiner Arbeit besser erhalten kann. Harrison und Vicente (1996) untersuchen den Einsatz solcher Menüs in Bezug auf die Lesbarkeit der Menüeinträge. Dabei vergleichen sie den Einsatz herkömmlicher Schrift mit sogenannten «Anti-Interference» Schriften – in Abbildung 28 rechts zu sehen – und können zeigen, dass diese einige Vorteile haben.

4.1.4. Marking Menüs – eine Weiterentwicklung der Pie Menüs

Das Konzept der Pie Menüs und deren Vorteile sind schon länger bekannt (siehe auch Seite 22). Eine Weiterentwicklung dieses Ansatzes sind die sogenannten Marking Menüs.

Mit Marking Menüs kann der Benutzer Menüpunkte auswählen, ohne dass er auf das Erscheinen des Menüs am Bildschirm warten muss. Wie in Abbildung 29 links zu sehen ist, kann ein Anfänger nach Aufrufen des Menüs warten, bis dieses am Bildschirm erscheint, den Menüpunkt für ein Untermenü wählen (Eintrag «Groceries») und nach dessen Erscheinen dort eine Option auswählen (Eintrag «Fruit & Veg»). Wie auf der rechten Seite der Abbildung 29 dargestellt, kann ein Benutzer mit der Zeit nur noch die Bewegung d.h. das Marking für den gewünschten Menüpunkt ausführen, um die entsprechende Auswahl zu treffen. Kurtenbach und Buxton (1993) haben die Vorteile und Grenzen solcher Marking Menüs untersucht.

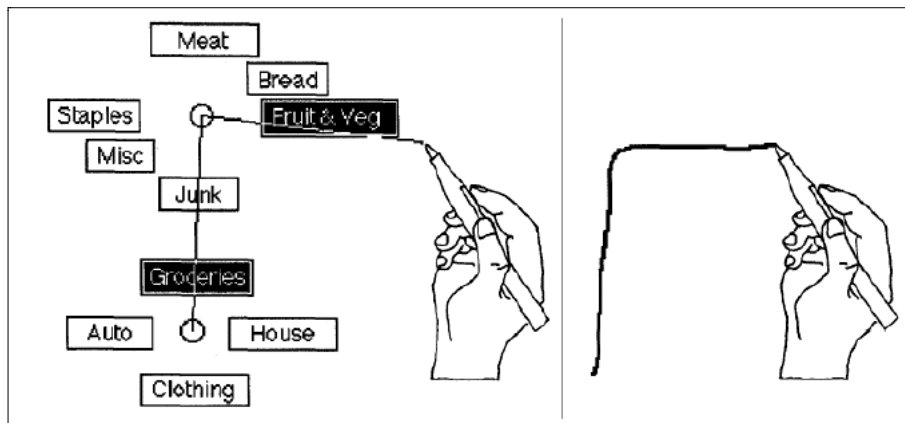


Abbildung 29 Marking Menü (aus Kurtenbach und Buxton, 1993)

Tapia und Kurtenbach (1995) schlagen mehrere Verbesserungen in Bezug auf Gestaltung und Verhalten der Marking Menüs vor. Speziell für die Eingabe mit einem Stift entwickelt, stellen Guimbretière und Winograd (2000) die sogenannten Flow Menüs vor.

4.2. «Menüs der Zukunft» in der Praxis – Maya von Alias|Wavefront

Maya ist eine professionelle 3-D Animationssoftware von Alias | Wavefront, mit welcher Hollywood-Filme wie «Jurassic Park» und «Toy Story» realisiert wurden.

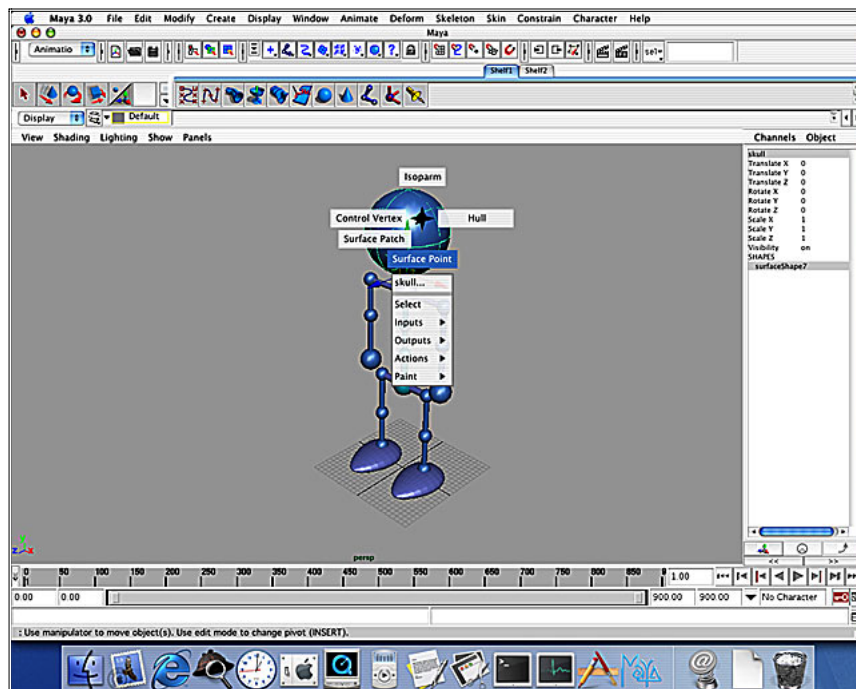


Abbildung 30 Maya User Interface (von www.aliaswavefront.com)

Das Erstellen von 3-D Animationen ist eine sehr komplexe Aufgabe, die Filmbranche ist jedoch stark geprägt von grossem Termindruck. Ein optimiertes User Interface bedeutet daher

einen Marktvorteil. Deshalb investiert Alias | Wavefront viel in die Erforschung neuer Interaktionstechniken und integriert laufend neuartige Menütechniken in ihr Produkt. Einige Mitglieder der Forschungsabteilung haben hier erwähnte Publikationen (mit-) verfasst: Bill Buxton (Chief Scientist), Gordon Kurtenbach (Director) und George Fitzmaurice (Research Scientist).

Im User Interface von Maya werden die verschiedensten Arten von Menüs eingesetzt, die zu grossen Teilen durch die Benutzer sehr stark individualisierbar sind. Einerseits verfügt Maya über eine reguläre Menüleiste – wie in Abbildung 30 zu sehen ist – und über andere von traditionellen GUIs bekannte Menüs. Andererseits sind in Maya einige weniger bekannte Menüarten wie die *Marking Menüs* zu finden, und mit der sogenannten «Hotbox» (Kurtenbach, Fitzmaurice, Owen und Baudel, 1999) verfügt Maya über ein spezielles Element, das schnellen Zugriff zu verschiedensten Menüs und so zu einer riesigen Anzahl von Optionen ermöglicht.

4.2.1. Marking Menüs

Wie auch viele andere weit verbreitete Programme bietet Maya die Möglichkeit, an den vom Benutzer erstellten Objekten Kontextmenüs (siehe auch Abbildung 11, Seite 14) aufzurufen.

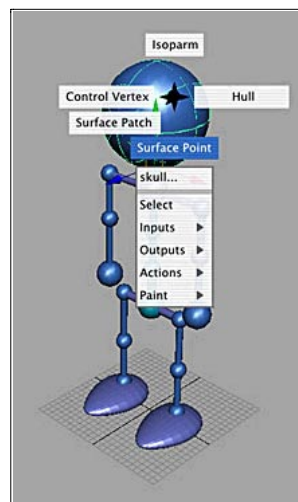


Abbildung 31 Objekt mit Marking Menü (von www.aliaswavefront.com)

In Maya sind solche Menüs aber auch in Form von Marking Menüs (siehe auch Abschnitt 4.1.4 auf Seite 29) realisiert, und wie von Tapia und Kurtenbach (1995) beschrieben, optimiert worden. Das Beispiel eines solchen Marking Menüs ist in Abbildung 31 zu sehen. Marking Menüs sind auch in der weiter unten beschriebenen «Hotbox» zu finden.

4.2.2. Die Hotbox

Kurtenbach et al. (1999) stellen ein speziell für Maya entwickeltes Interfacelement vor: die *Hotbox*. Die Abbildung 32 zeigt das User Interface im Überblick mit der aufgerufenen Hotbox.

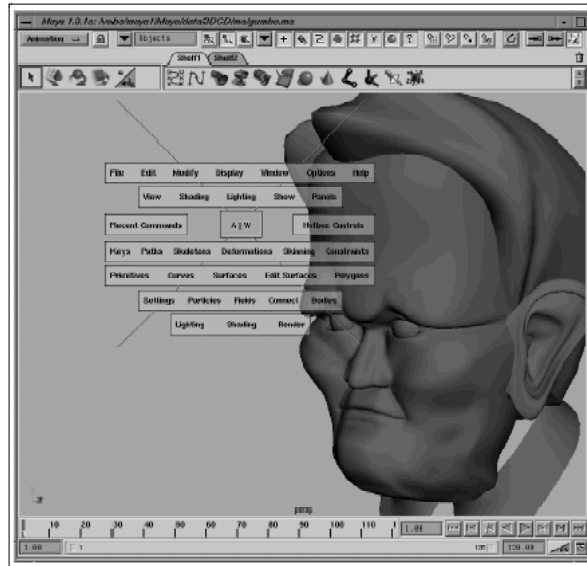


Abbildung 32 Maya Hotbox (aus Kurtenbach et al., 1999)

Da Maya gesamthaft über etwa 1'200 Befehle verfügt, und die Benutzer viele dieser Befehle möglichst schnell im Kontext ihrer Arbeit zur Verfügung haben sollten, mussten die Forscher von Alias | Wavefront dafür ein Hilfsmittel entwickeln. Die Lösung kombiniert in neuartiger Weise bereits bekannte Techniken.

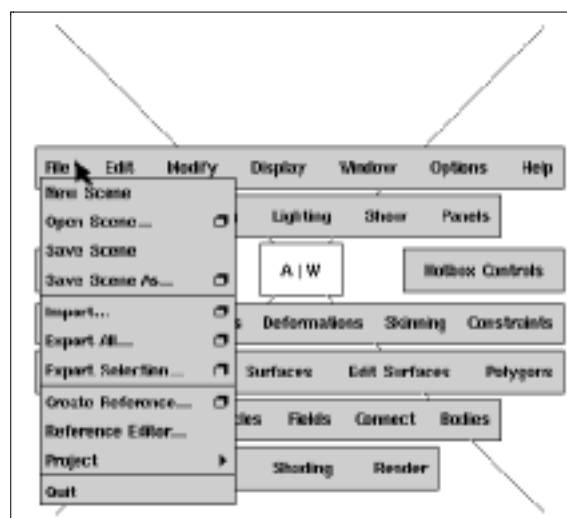


Abbildung 33 Lineares Menü in der Hotbox (aus Kurtenbach et al., 1999)

Durch das Drücken der Leertaste kann der Benutzer die Hotbox aufrufen, die daraufhin zentriert an der aktuellen Cursorposition erscheint. Die Hotbox besteht unter anderem aus verschiedenen Reihen von «herkömmlichen» Menüleisten (z.B. «File, Edit, Modify ...»), aus welchen lineare Menüs aufgerufen werden können – ein Beispiel ist in Abbildung 33 zu sehen. Zudem sind auch Marking Menüs in den Menüleisten der Hotbox zu finden – ein entsprechendes Beispiel ist in Abbildung 34 zu sehen.

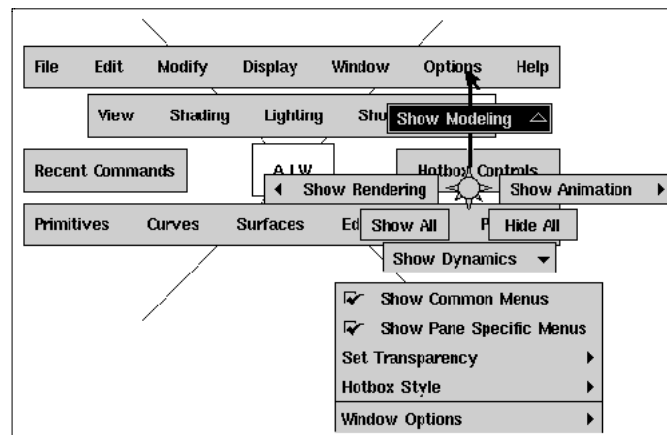


Abbildung 34 Marking Menü in der Hotbox (aus Kurtenbach et al., 1999)

Eine der Menüleisten enthält ein Marking Menü zum Konfigurieren der Hotbox. Dieses Menü ist in Abbildung 34 zu sehen. Hier lässt sich unter anderem auch die Transparenz der Hotbox einstellen – die Menüeinträge werden dabei in einer speziellen Schrift dargestellt (siehe auch Abschnitt 4.1.3, Seite 28).

Unter den Menüleisten befinden sich zusätzlich noch fünf Zonen, die es ermöglichen, benutzerdefinierte Menüs aufzurufen. Eine der Zonen ist das Rechteck in der Mitte («Center Zone <A | W>»⁵). Die anderen Zonen werden durch das Kreuz (X) definiert («North, East, South und West Zone»). Durch das Drücken einer Maustaste in einer der Zonen kann je ein anderes Marking Menü aufgerufen werden. Mittels unterschiedlicher Belegung von drei Maustasten können so durch die Benutzer – je nach Bedarf – 15 verschiedene Menüs definiert werden. Dadurch, dass zur «Center Zone» hin keine Entfernung zurückgelegt werden muss und die anderen Zonen eine maximale Größe haben, sind die Menüs sehr schnell zu erreichen. Die Erkenntnisse aus Fitts's Law (siehe Abschnitt 3.2.1, Seite 19) werden dadurch optimal ausgenutzt.

⁵ Das «A | W» hat keine Bedeutung sondern ist lediglich ein Symbol für den Hersteller von Maya: Alias | Wavefront.

5. Diskussion

Menüs sind zweifellos ein sehr nützliches Konzept mit vielen guten Eigenschaften. Sie haben auch heute nicht an Bedeutung verloren. Deshalb wird stets zu diesem Thema Forschung betrieben, und manches Forschungsergebnis konnte so zur Verbesserung von User Interfaces bei Computern beitragen. Aber wie auch in anderen HCI Gebieten ist das optimale Verfügbarmachen dieses Wissens für die Praxis ein noch ungelöstes Problem, und daher werden auch «alte Fehler» immer wieder neu gemacht.

Zwar scheinen zu einigen Fragen der Strukturierung und Präsentation von Menüs die Forschungsbemühungen abgeschlossen zu sein. Trotzdem ergeben sich in beiden Bereichen durch die Entwicklung immer wieder neue Fragen. Deshalb ist auch ein Ende der Menüforschung nicht abzusehen.

5.1. Strukturierung von Menüs

In den frühen Phasen der Menüforschung wurden viele Untersuchungen zur Strukturierung von Menüs durchgeführt. Viele dieser Erkenntnisse sind dann auch in plattformspezifische Guidelines (z.B. Apple Computer, Inc., 1995) eingeflossen oder in Form von Büchern zu Interface Design (z.B. Mayhew, 1992) publiziert worden.

Da die Anpassung einer Menüstruktur an die Charakteristiken einer Aufgabe sehr wichtig ist (siehe dazu z.B. Norman, 1991 oder Mayhew, 1992), wurde erkannt, dass auch die Weiterentwicklung der Methoden für eine benutzerzentrierte Softwareentwicklung notwendig ist. Diese beinhalten die Untersuchung der zu unterstützenden Aufgaben mittels Aufgabenanalysen (Hackos und Redish, 1998) – als Grundlage für eine sinnvolle Menüstruktur – und die Realisierung in einem iterativen Usability Engineering Prozess (Mayhew, 1999). Nur mit solchen Methoden kann sichergestellt werden, dass die Struktur von Menüs optimal auf die jeweiligen Benutzer einer Software und die damit zu erledigenden Aufgabe abgestimmt ist.

Immer wieder wurden jedoch auch neue Menüs entwickelt, welche Besonderheiten aufwiesen, die dann mit herkömmlichen Lösungen verglichen werden mussten. Ein Beispiel für eine neuere Entwicklung sind auch die sogenannten «Adaptive Menüs», wie sie in Windows 2000 zu finden sind. Zwar gibt es zu dieser Art Menüs noch keine publizierten Studien⁶, aber zum Beispiel Raskin (2000, S. 70) äussert sich eher skeptisch zu solchen Mechanismen.

⁶ Zwar haben Mitchell und Shneiderman (1989) eine Studie zu «Adaptive dynamic menus» durchgeführt, diese Menüs zeigten aber ein anderes Verhalten als die hier erwähnten.

Mit der Entwicklung hin zu Menüs, die immer mehr Menüpunkte enthalten, bleibt die Frage nach den verwendeten Suchstrategien innerhalb von Menüs interessant. Die jüngsten Studien mittel kognitiver Modellierung und Eytracking (siehe Abschnitt 4.1.1, Seite 23) zeigen, dass in dieser Frage immer noch aktiv geforscht wird.

5.2. Präsentation von Menüs

Die Optimierung der Präsentation von lange bekannten Menüarten wie den Pull-down Menüs hat sich bereits weitgehend etabliert, obwohl beim Betrachten einiger Web Interfaces der Verdacht aufkommt, dass dieses Wissen doch noch nicht weit genug verbreitet ist. Die Entwickler von Software für GUIs haben hier den Vorteil, dass sie meist auf bereits vorgefertigte Komponenten zurückgreifen können, die über längere Zeit hin optimiert wurden.

Das Gebiet der Information Visualization (Abschnitt 4.1.2.3, Seite 26), welches eines der aktivsten HCI Forschungsgebiete ist, wird wahrscheinlich noch viele neuartige Lösungen entwickeln. Auch die bestehenden Umsetzungen werden immer weiter verbessert und auch kommerziell angeboten.

Die Entwicklung ganz spezifischer Lösungen für einzelne Produkte wie z.B. Maya (Abschnitt 4.2, Seite 30) wird bestimmt auch in Zukunft interessant zu beobachten sein, weil sich dort die Menüforschung mit konkreter benutzerzentrierter Softwareentwicklung aus der Praxis trifft.

Literaturverzeichnis

- Aaltonen, A., Hyrskykari, H. & Rähkä, K.-J. (1998). 101 Spots, or How Do Users Read Menus? *Conference proceedings on Human factors in computing systems CHI'98* (pp. 132–139). New York, NY: ACM.
- Apple Computer, Inc. (1987). *Human Interface Guidelines: The Apple Desktop Interface*. Reading, MA: Addison-Wesley Publishing Company.
- Apple Computer, Inc. (1995). *Macintosh Human Interface Guidelines* [On-line]. Available: <http://developer.apple.com/techpubs/mac/HIGuidelines/HIGuidelines-2.html>
- Apple Computer, Inc. (1997). *Mac OS 8 Human Interface Guidelines* [On-line]. Available: <http://developer.apple.com/techpubs/mac/HIGOS8Guide/thig-2.html>
- Apple Computer, Inc. (2001). *Aqua Human Interface Guidelines* [On-line]. Available: <http://developer.apple.com/techpubs/macosx/Essentials/AquaHIGuidelines>
- Bederson, B. B. (2000). Fisheye menus. *Proceedings of the 13th annual ACM symposium on User interface software and technology UIST'00* (pp. 217–225). New York, NY: ACM.
- Bier, E. A., Stone, M. C., Pier, K., Buxton, W. & DeRose, T. D. (1993). Toolglass and Magic Lenses: the See Throughh Interface. *Proceedings of Siggraph 93 – Computer Graphics Annual Conference Series* (pp. 73–80). New York, NY: ACM.
- Byrne, M. D., Anderson, J. R., Douglass, S. & Matessa, M. (1999). Eye tracking the visual search of click-down menus. *Conference proceedings on Human factors in computing systems CHI'99* (pp. 402–409). New York, NY: ACM.
- Callahan, J., Hopkins, D., Weiser, M. & Shneiderman, B. (1988). An empirical comparison of pie vs. linear menus. *Conference proceedings on Human factors in computing systems CHI'88* (pp. 95–100). New York, NY: ACM.
- Card, S. K., Mackinlay, J. D. & Shneiderman, B. (Eds.). (1999). *Readings in Information Visualization*. San Francisco, CA: Morgan Kaufman Publishers, Inc.
- Card, S. K., Moran, T. P. & Newell, A. (1983). *The Psychology of Human-Computer Interaction*. Hillsdale, NJ: Lawrence Erlbaum Associates, Inc., Publishers.
- Cooper, A. (1995). *About Face: The Essential of User Interface Design*. Foster City, CA: IDG Books Worldwide, Inc.
- Fleming, J. (1998). *Web Navigation: Designing the User Experience*. Sebastopol, CA: O'Reilly & Associates, Inc.
- Furnas, G. W. (1986). Generalized fisheye views. *Conference proceedings on Human factors in computing systems CHI'86* (pp. 16–23). New York, NY: ACM.

- Guimbretière, F. & Winograd, T. (2000). FlowMenu: combining command, text, and data entry. *Proceedings of the 13th annual ACM symposium on User interface software and technology UIST'00* (pp. 213–216). New York, NY: ACM.
- Hackos, J. T. & Redish J. C. (1998). *User and Task Analysis for Interface Design*. New York, NY: John Wiley & Sons, Inc.
- Harrison, B. L. & Vicente, K. J. (1996). An experimental evaluation of transparent menu usage. *Conference proceedings on Human factors in computing systems CHI'96* (pp. 391–398). New York, NY: ACM.
- Herman, I., Melançon, G. & Marshall, M. S. (2000). Graph Visualization and Navigation in Information Visualization: a Survey. *IEEE Transactions on Visualization and Computer Graphics*, 6 (1), 24–43.
- Hornof, A. J. & Kieras, D. E. (1997). Cognitive modeling reveals menu search in both random and systematic. *Conference proceedings on Human factors in computing systems CHI'97* (pp. 107–114). New York, NY: ACM.
- Hornof, A. J. & Kieras, D. E. (1999). Cognitive modeling demonstrates how people use anticipated location knowledge of menu items. *Conference proceedings on Human factors in computing systems CHI'99* (pp. 410–417). New York, NY: ACM.
- Hutchins, E. L., Hollan, J. D. & Norman, D. A. (1986). Direct Manipulation Interfaces. In D. A. Norman & S. W. Draper (Eds.), *User Centered System Design* (pp. 87–124). Hillsdale, NJ: Lawrence Erlbaum Associates, Inc., Publishers.
- International Organization for Standardization (ISO). (1996). *ISO 9241-10. Ergonomische Anforderungen für Bürotätigkeiten mit Bildschirmgeräten – Teil 10: Grundsätze der Dialoggestaltung*. Brüssel, BE: CEN.
- Johnson, B. & Shneiderman, B. (1991). Treemaps: a space-filling approach to the visualization of hierarchical information structures. *Proceedings of the 2nd International IEEE Visualization Conference* (pp. 275-291). San Diego, CA: IEEE CS Press.
- Johnson, J. (2000). *GUI Bloopers – Dont's and Do's for Software Developers and Web Designers*. San Diego, CA: Academic Press, Inc.
- Johnson, J., Roberts, T. L., Verplank, W., Smith, D. C., Irby, C. H., Beard, M. & Mackey, K. (1995). The Xerox Star: A Retrospective. In R. M. Baecker, J. Grudin, W. A. Buxton & S. Greenberg (Eds.), *Readings in Human-Computer Interaction: Toward the Year 2000* (2nd ed., pp. 53–70). San Francisco, CA: Morgan Kaufman Publishers, Inc.
- Kurtenbach, G. & Buxton, W. (1993). The Limits Of Expert Performance Using Hierarchic Marking Menus. *Conference proceedings on Human factors in computing systems CHI'93* (pp. 482–487). New York, NY: ACM.

- Kurtenbach, G., Fitzmaurice, G. W., Owen, R. N. & Baudel, T. (1999). The Hotbox: efficient access to a large number of menu-items. *Conference proceedings on Human factors in computing systems CHI'99* (pp. 231–237). New York, NY: ACM.
- Lynch, P. J. & Horton, S. (1999). *Web Style Guide*. New Haven: Yale University Press.
- Mandel, T. (1997). *The Elements of User Interface Design*. New York, NY: John Wiley & Sons, Inc.
- Mayhew, D. (1992). *Principles and Guidelines in Software User Interface Design*. Englewood Cliffs, NJ: Prentice Hall.
- Mayhew, D. (1999). *The Usability Engineering Lifecycle*. San Francisco, CA: Morgan Kaufmann Publishers.
- Microsoft Corp. (1999). *The Microsoft Windows User Experience*. Redmond, WA: Microsoft Press.
- Mills, Z. & Prime, M. (1990). Are all menus the same? – An empirical study. *Conference proceedings on Human-Computer Interaction – INTERACT 90* (pp. 423–427). Amsterdam: Elsevier Science Publishers B. V. (North-Holland).
- Mitchell, J. & Shneiderman, B. (1989). Dynamic versus static menus: an exploratory comparison. *SIGCHI Bulletin*, 20 (4), 33–37.
- Nielsen, J. (1993). *Usability Engineering*. Chestnut Hill, MA: Academic Press.
- Nielsen, J. (1999). *Designing Web Usability: The Practice of Simplicity*. Indianapolis, IN: New Riders Publishing.
- Norman, K. L. (1991). *The psychology of menu selection: Designing cognitive control at the human/computer interface*. Norwood, N.J.: Ablex Publishing Corporation.⁷
- Paap, K. R. & Cooke, N. J. (1997). Design of Menus. In M. H. Helander, T. K. Landauer & P. V. Prabhu (Eds.), *Handbook of Human-Computer Interaction* (2nd ed., pp. 533–572). Amsterdam, NL: Elsevier Science B.V.
- Plaisant, C., Milash, B., Rose, A., Widoff, S. & Shneiderman, B. (1996). LifeLines: visualizing personal histories. *Conference proceedings on Human factors in computing systems CHI'96* (pp. 221–227). New York, NY: ACM.
- Rao, R. & Card, S. K. (1994). The Table Lens: Merging Graphical and Symbolic Representations in an Interactive Focus + Context Visualization for Tabular Information. *Conference proceedings on Human factors in computing systems CHI'94* (pp. 318–482). New York, NY: ACM.

⁷ Das ganze Buch von Norman (1991) ist auch on-line verfügbar:
<http://lap.umd.edu/pomsFolder/pomsHome.html>

- Raskin, J. (2000). *The Humane Interface*. Reading, MA: Addison-Wesley Longman, Inc.
- Richter, M. (1997). *Kriterien der Benutzerfreundlichkeit*. Unveröff. Literaturarbeit, Universität Zürich, Psychologisches Institut, Abt. Angewandte Psychologie.
- Robertson, G. G., Mackinlay, J. D. & Card, S. K. (1991). Cone Trees: animated 3D visualizations of hierarchical information. *Conference proceedings on Human factors in computing systems CHI'91* (pp. 189–194). New York, NY: ACM.
- Rosenfeld, L. & Morville, P. (1998). *Information Architecture for the World Wide Web*. Sebastopol, CA: O'Reilly & Associates.
- Sears, A. & Shneiderman, B. (1994). Split Menus: Effectively Using Selection Frequency to Organize Menus. *ACM Transactions on Computer-Human Interaction*, 1 (1), 27–51.
- Shackel, B. (1991). Usability – Context, Framework, Definition, Design and Evaluation. In B. Shackel & S. Richardson, (Eds.), *Human Factors for Informatics Usability* (pp. 21–37). Cambridge, UK: University Press.
- Shneiderman, B. (1998). *Designing the User Interface* (3rd ed.). Reading, MA: Addison-Wesley Longman, Inc.
- SIGCHI – ACM Special Interest Group on Computer-Human Interaction (2001). *Curricula for Human-Computer Interaction* [On-line]. Available: <http://www.sigchi.org/cdg>
- Spool, J. M., Scanlon, T., Schroeder, W., Snyder, C. & DeAngelo, T. (1999). *Web Site Usability: a Designer's Guide*. San Francisco, CA: Morgan Kaufman Publishers, Inc.
- Sun Microsystems, Inc. (2001). *Java look and feel design guidelines* (2nd ed.) [On-line]. Available: <http://java.sun.com/products/jlf/ed2/book/index.html>
- Tapia, M. A. & Kurtenbach, G. (1995). Some design refinements and principles on the appearance and behavior of marking menus. *Proceedings of the 8th annual ACM symposium on User interface software and technology UIST'95* (pp. 189–195). New York, NY: ACM.
- Tognazzini, B. (1993). *Tog on Interface*. Reading, MA: Addison-Wesley Publishing Company.
- Tognazzini, B. (1999). *A Quiz Designed to Give You Fitts* [On-line]. Available: <http://www.asktog.com/columns/022DesignedToGiveFitts.html>
- Walker, N. & Smelcer, J. B. (1990). A Comparison of Selection Times from Walking and Pull-Down Menus. *Conference proceedings on Human factors in computing systems CHI'90* (pp. 221–225). New York, NY: ACM.