

## GRUNDDIMENSIONEN VON INTERAKTIONSFORMEN

Jürgen E. Ziegler, Stuttgart

**Zusammenfassung:** Für software-ergonomische Untersuchungen und Gestaltungen ist es wesentlich, die Art der Interaktion, die vom System angeboten wird, möglichst objektiv und eindeutig beschreiben zu können. Rein deskriptive oder eindimensionale Klassifizierungen sind hierfür nicht ausreichend. In diesem Beitrag werden deshalb drei Grunddimensionen von Interaktionsformen aus einem Modell der Mensch-Rechner-Interaktion abgeleitet und unterschiedliche Ausprägungen auf diesen Dimensionen definiert. Die Einordnung von Interaktionsformen in diesen Rahmen wird am Beispiel der "direkten Manipulation" diskutiert.

### Einleitung

Seit der Auflistung verschiedener Dialogtechniken durch Martin (1973) und seiner Aufteilung nach benutzer- und computer-initiierten Techniken sind verschiedene Versuche gemacht worden, die Vielfalt verschiedener Interaktionsformen zu erfassen, zu beschreiben und zu strukturieren. Hierbei wurden verschiedenartige Strukturierungsmerkmale verwendet, die sich z. B. am verwendeten Eingabegerät, der Art der Initiierung von Dialogschritten, synchronen oder asynchronen Interaktionen u.a. orientieren. Meist wird jedoch nur eine Aufzählung der als relevant betrachteten Interaktionstechniken mit Beschreibung der jeweiligen typischen Eigenschaften geliefert (vergl. z.B. Shneiderman, 1987).

Unterscheidungen wie computer- versus benutzerinitiiert orientieren sich an einem einzelnen Merkmal, das zur Einordnung herangezogen wird. Moderne, hoch interaktive Benutzerschnittstellen (z.B. graphische, objektorientierte Schnittstellen) machen solche Unterscheidungen fragwürdig, da gleichzeitig sowohl Aspekte der Benutzer- wie der Rechner-Initiierung anzutreffen sind. Darüber hinaus sind die meisten realen Systeme nicht durchgängig nach einem einzigen Interaktionsprinzip aufgebaut, sondern verwenden verschiedenartige Techniken gleichzeitig. Ebenso sind unterschiedlichste Mischformen zu beobachten, die zu einer Überlappung von Techniken führt. Aus diesen Gründen erscheint eine rein deskriptive oder eindimensionale Einordnung von Interaktionstechniken nicht ausreichend.

Die angeführte Vielfalt und Komplexität der Eigenschaften interaktiver Systeme führt zu verschiedenen Problemen bei software-ergonomischen Untersuchungen und Gestaltungen:

- die Verwendung bislang schlecht definierter Begriffe wie etwa "direkte Manipulation" erschwert ein gemeinsames Verständnis der damit bezeichneten Systemeigenschaften;
- die Generalisierbarkeit von Ergebnissen aus Benutzbarkeitsstudien ist eingeschränkt;
- für vergleichende Untersuchungen von Systemen mit unterschiedlichen Interaktionstechniken müssen diese Techniken objektiv erfaßbar sein.

Wesentlich für eine objektive Erfassung von Interaktionsformen ist es, daß diese als Eigenschaft des Systems klar von psychologisch oder ergonomisch ausgerichteten Benutzungseigenschaften getrennt werden können. Der vorliegende Beitrag beschreibt einen Ansatz mit drei Dimensionen, die zur Klassifikation der Eigenschaften eines interaktiven Systems herangezogen werden können. Die Kombination der Ausprägungen auf den drei Achsen bestimmt den jeweiligen Typ einer Interaktionstechnik. Dabei werden die Dimensionen aus einem allgemeinen Modell der Mensch-Rechner-Interaktion abgeleitet.

#### Ableitung von Grunddimensionen aus einem Modell interaktiver Systeme

In diesem Abschnitt soll dargestellt werden, wie Strukturierungsdimensionen für Interaktionstechniken aus einem Modell des interaktiven Systems abgeleitet werden können. Dazu soll zunächst das Gesamtsystem Benutzer-Rechner betrachtet werden, um anschließend Komponenten und Struktur auf der Rechnerseite als Untermodell hierzu näher beschreiben zu können.

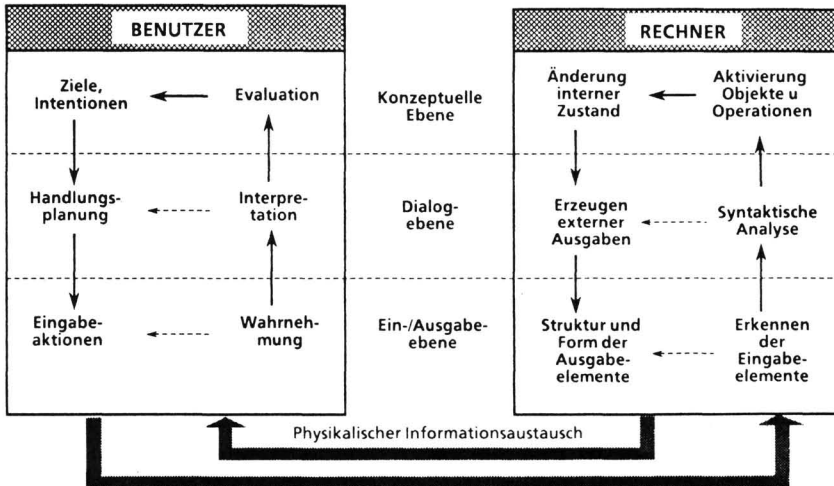
Als Ausgangspunkt dient ein Kommunikationsmodell der Mensch-Rechner Interaktion, bei dem der Kommunikationsprozeß auf unterschiedlichen Abstraktionsebenen abläuft. Solche Schichtenmodelle wurden - allerdings mit unterschiedlicher Anzahl von Ebenen - häufig zur Beschreibung von Benutzerschnittstellen herangezogen (vgl. Moran, 1981; Fähnrich & Ziegler, 1984; Nielsen, 1986). Hier sollen in Anlehnung an Moran (1981) drei Hauptebenen unterschieden werden:

- Konzeptuelle Ebene: beschreibt Aufgaben, die mit dem System durchgeführt werden können, sowie die dabei verwendeten konzeptuellen Objekte und Operationen.
- Dialogebene: beschreibt Dialogstrukturen und -abläufe und den syntaktischen Aufbau von Ein- und Ausgaben.
- Ein-/Ausgabeebene: beschreibt Struktur (z.B. zweidimensionale Anordnung) und Form der Elemente der Ein-/Ausgabe.

Auf eine mögliche weitere Untergliederung dieser Ebenen soll hier nicht eingegangen werden. Da der reale Informationsaustausch nur physikalisch erfolgen kann, ist es sinnvoll, Prozesse anzunehmen, die sowohl auf der Seite des Be-

nutzers wie auch des Rechners in Eingabe- und Ausgaberrichtung die erforderlichen Transformationen zwischen den Ebenen durchführen. Damit gelangt man zu einem geschlossenen Interaktionszyklus, wie er in Abb. 1 dargestellt ist. Entsprechend diesem Modell sollen folgende Begriffe definiert werden: Kommunikation bezeichnet den (virtuellen) Informationsfluß auf den unterschiedlichen Ebenen; Interaktion ist die (in sich geschlossene) Abfolge der beteiligten Prozesse. Die Begriffe beschreiben also das gleiche Gesamtphänomen aus verschiedenen Blickwinkeln.

Abb. 1: Interaktionszyklus in einem Modell der Mensch-Rechner-Interaktion



Aus diesem Interaktionsmodell soll nun die rechnerseitige Komponente herausgegriffen und detaillierter beschrieben werden, um damit mögliche Interaktionsformen zu klassifizieren.

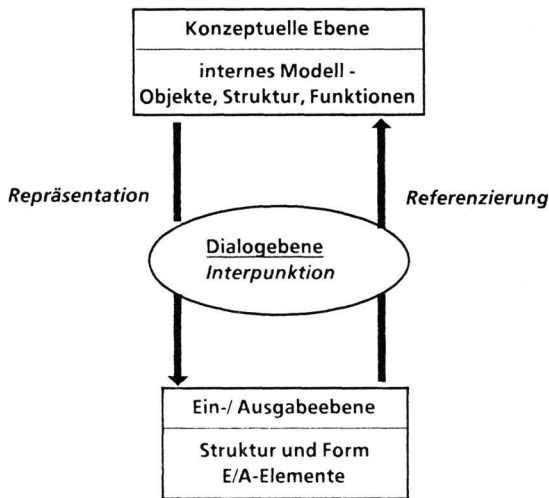
Auf der obersten Ebene bestimmt das vom Systemdesigner entworfene konzeptuelle Modell die Objekte und Funktionen für den jeweiligen Anwendungsbereich. Dieses konzeptuelle Modell ist für den Benutzer nicht direkt zugänglich und kann deshalb als internes Modell bezeichnet werden. Das konzeptuelle Modell kann weiter zergliedert werden in Objekte mit Attributwerten, Relationen zwischen Objekten und die anwendbaren Operationen. Um dem Benutzer Informationen über den internen Systemzustand zur Verfügung zu stellen, wird ein Ausschnitt des Zustandes des internen Modells an der sichtbaren Oberfläche (Bildschirm) abgebildet. Diese Umsetzung von internem Zustand auf die externe Darstellung soll als Repräsentation bezeichnet werden.

In der Eingaberichtung muß der Benutzer Objekte und Funktionen des internen Modells ansprechen können, um entsprechende Zustandsänderungen zu steuern. Die Abbildung von Eingabeelementen auf interne Objekte soll als Referenzierung bezeichnet werden.

Schließlich muß berücksichtigt werden, daß das Erzeugen von Eingabeobjekten durch den Benutzer nicht unmittelbar zu internen Zustandsänderungen führen muß, sondern in Abhängigkeit von der Syntax mehr oder weniger komplexe Eingaben formuliert werden können, die erst nach einer Terminierung verarbeitet werden. Dieser Aspekt soll als Interpunktion (oder Interaktivität) bezeichnet werden.

Die drei dargestellten Begriffe Repräsentation, Referenzierung und Interpunktion beschreiben grundlegende Eigenschaften interaktiver Systeme und können zu einer Klassifikation von Interaktionsformen herangezogen werden (Ziegler, Hoppe, Eichhorn & Puetz, 1986). Abb. 2 zeigt das Verhältnis der Dimensionen zueinander sowie zu den Ebenen des Interaktionsmodells.

Abb. 2: Grunddimensionen im Interaktionsmodell



#### Repräsentation

Ein wesentliches Unterscheidungsmerkmal von Benutzerschnittstellen ist die Art der Abbildung von internem Systemzustand auf die sichtbare Oberfläche. Im allgemeinen Fall ist die externe Darstellung eine Überlagerung mehrerer Komponenten, deren Inhalt abhängig ist vom aktuellen internen Zustand, einer Anzahl vorangegangener Zustände (z.B. Ausgaben von früheren Interaktionsschritten in einem Teletype-Fenster) sowie den aktuellen und früheren Benutzereingaben (als "Echo" dieser Eingabe).

Für die Art der Repräsentationsabbildung können nun zwei Unterscheidungen getroffen werden: Im ersten Fall ist die externe Darstellung nur von vorangegangenen internen Zuständen abhängig, aber nicht vom aktuellen Zustand. Nachdem ein Ausgabeobjekt erzeugt worden ist, gibt es keine Rückverweisungsmöglichkeit von diesem Objekt auf den aktuellen Zustand. Dies bedeutet, daß der Benutzer nicht auf das dargestellte Objekt zugreifen, es aktivieren oder verändern kann. Die einzige Möglichkeit zur Veränderung der Darstellung besteht im Erzeugen neuer Ausgabeobjekte durch das System. Mallgren (1983) nennt diesen Typ von externen Objekten "sequential data type". Er soll hier als sequentielle Repräsentation bezeichnet werden. Beispiele für diesen Typ finden sich bei zeilenorientierten Kommandosprachen für Betriebssysteme, Anfragesprachen u. a..

Im zweiten Fall besitzen die extern dargestellten Objekte einen aktuellen internen Zustand und können deshalb vom Benutzer und vom System angesprochen und verändert werden. Der Benutzer kann in diesem Fall durch Veränderung von externen Objekten der Ausgabe den internen Systemzustand beeinflussen. In diesem Fall können die dargestellten Objekte als "shared data type", der Repräsentationstyp als parallele Repräsentation bezeichnet werden. Beispiele dafür sind etwa das Verändern eines Feldes in einem Formular oder die Manipulation von Ikonen am Bildschirm.

Der zweite Fall (shared data type) kann noch weiter unterschieden werden, indem die verschiedenen Komponenten des internen und externen Modells in Betracht gezogen werden. Im ersten Unterfall liegt eine umkehrbare Repräsentationsabbildung nur für Objektattribute vor, die Struktur der Objekte wird invariabler Form, z.B. als zweidimensionale Anordnung dargestellt. Formulare sind Beispiele für diesen Fall, da eine Änderung von Feldwerten zwar den internen Attributwert ändern kann, die Struktur des Formulars aber fest vorgegeben ist. Im zweiten Fall existiert eine Rückabbildung auch von der Struktur der externen Objekte auf den internen Zustand. Beispiel dafür ist das Bewegen eines Dokumentikons auf ein Ablageikon, wodurch auch intern das Dokument an anderer Stelle in der Ablagehierarchie eingefügt wird.

Mit diesen Unterscheidungen lassen sich drei Repräsentationstypen beschreiben:

1. Sequentielle Repräsentation
2. Parallele Repräsentation von Objekten
3. Parallele Repräsentation von Objekten und Objektstruktur

#### Referenzierung

In der Eingaberichtung muß der Benutzer Objekte und Operationen des konzeptuellen Modells durch geeignete Eingaben ansprechen können, um die

gewünschten (benutzergesteuerten) Veränderungen des internen Zustandes zu bewirken. Dabei werden die vom Benutzer verwendeten Eingabeobjekte auf Objekte des internen Modells abgebildet. Für diese Abbildung soll der Begriff Referenzierung verwendet werden.

Bei der Referenzierung können ebenfalls drei Grundtypen unterschieden werden, die die unterschiedlichen Zugriffsmöglichkeiten des Beutzers auf die internen Objekte beschreiben:

1. Referenz durch eine Zeigeoperation auf ein Objekt der externen Darstellung (deiktische Referenz). Die Selektion eines externen Objekts kann dann vom System auf ein internes Objekt abgebildet werden. Da in diesem Fall Elemente der Ausgabe direkt wieder als Eingaben verwendet werden können, wird auch von interreferentieller Ein- / Ausgabe gesprochen (Draper, 1986). Deiktische Referenz ist nur bei den Repräsentationstypen 2 und 3 möglich, da nur hier eine Rückabbildung von externen auf interne Objekte möglich ist.
2. Referenz durch Namen. Hierbei generiert der Benutzer ein Eingabesymbol, das das entsprechende interne Objekt bzw. die Operation identifiziert. Bei diesem Typ wird kein direkter Bezug auf ein externes Objekt genommen.
3. Referenz durch Beschreibung (deskriptive Referenz): Objekte bzw. Objektmengen können auch dadurch identifiziert werden, daß ihre Attributwerte spezifiziert werden. Meist ist in die Beschreibung der Name der betreffenden Objektklasse eingeschlossen. Ein typisches Beispiel für diesen Fall sind Abfragesprachen (z.B. "zeige die Namen aller Angestellten, deren Gehalt höher als x ist.").

Die dargestellten Referenztypen kennzeichnen unterschiedliche Stufen der Indirektheit, mit der interne Objekte angesprochen werden können. Hier sind unterschiedliche Mischungen von Indirektheit möglich (z.B. Zeigen auf Attributwerte, die ein Objekt bestimmen), wobei die hier dargestellten Fälle als Grundformen angesehen werden können.

### Interpunktion

Der Benutzer kann bei unterschiedlichen Schnittstellen syntaktisch mehr oder weniger komplexe Eingaben formulieren, bevor diese vom System interpretiert und in eine interne Zustandsänderung umgesetzt werden. Interpunktion beschreibt die zeitliche Segmentierung des Informationsaustauschs zwischen Benutzer und System. Sie gibt also verschiedene Stufen der Interaktivität an. Auch hier sollen drei Grundtypen unterschieden werden:

1. Funktionseingabe ohne Parameter: Hierbei wird jeweils eine einzelne Funktion eingegeben und sofort vom System interpretiert und entsprechendes Feedback geliefert. Beispiele sind etwa Selektionen mit der Maus oder das Löschen eines bereits aktivierten Objektes.
2. Funktionseingaben mit Spezifikation eines oder mehrerer Parameter: Dies ist typischerweise der Fall bei Kommandosprachen (z.B. "lösche File1.") Eingaben dieses Typs erfordern normalerweise einen Abschluß durch ein geeignetes Terminierungssymbol.
3. Eingaben mit Spezifikation mehrerer Funktionen und mehrerer Parameter: Dieser Fall kennzeichnet die Spannweite von komplexen Kommandosprachen (z.B. Eingabe einer Pipe in UNIX) bis hin zu Programmiersprachen bei Hinzunahme von Kontrollstrukturen.

Interpunktion beschreibt in gewissem Sinne die Mächtigkeit einer Benutzereingabe, die zunächst vollständig formuliert wird, bevor das System die entsprechenden Operationen ausführt.

#### Bestimmung von Interaktionsformen am Beispiel "Direkte Manipulation"

Direkte Manipulation (DM) ist ein im Bereich der Mensch-Rechner-Interaktion vieldiskutierter, aber nichtsdestotrotz nur vage definierter Begriff. Beispiele finden sich bei Systemen mit Schreibtischoberfläche (Desktop-Metapher), auf der Dokumente, Ordner und andere Objekte als Ikonen manipuliert werden können, bei WYSIWYG-Editoren (What you see is what you get) u.a..

Shneiderman (1982), der den Begriff DM eingeführt hat, charakterisiert die Interaktionsform durch drei Hauptmerkmale: kontinuierliche Objektrepräsentation; physische Interaktion anstelle komplexer Kommandos; schnelle, inkrementelle und reversible Aktionen mit unmittelbarem Feedback. Hutchins, Hollan & Norman (1986) charakterisieren DM über den Begriff der Direktheit und geben an, daß bei DM der semantische Abstand zwischen den Objekten der Aufgabe und den Rechnerobjekten verringert ist.

Beide Definitionen zeigen zwar typische System- oder Benutzungseigenschaften von DM, machen es aber nicht möglich, klar anzugeben, ob eine Schnittstelle bzw. ein Teil der Schnittstelle in diese Klasse fällt.

Im Rahmen der vorgestellten Grunddimensionen läßt sich nun eine genauere Einordnung der Phänomene bei DM vornehmen. Hierbei sollen die Eigenschaften von typischen Systemen dieser Klasse (z.B. Xerox Star oder Apple Macintosh) betrachtet werden

Eine notwendige Eigenschaft von DM-Systemen besteht darin, daß der Benutzer durch Manipulation von Oberflächenobjekten den internen Zustand be-

einfließen kann. Deshalb ist zumindest eine Repräsentation vom Typ 2 (parallele Repräsentation) notwendig. Für Anwendungen, die z.B. mit der Desktop-Metapher operieren, ist Typ 3 erforderlich, da strukturelle Veränderungen an der Oberfläche auch zu internen strukturellen Änderungen führen müssen.

Eine ebenfalls notwendige Anforderung ist die Verfügbarkeit von Zeigoperationen (z.B. mit der Maus), mit deren Hilfe Objekte und Operationen selektiert werden können (deiktische Referenz). Daneben treten in realen Systemen aber auch Referenzen über Namen auf. Insbesondere bei Funktionen muß berücksichtigt werden, daß diese nicht als Objekte im eigentlichen Sinn dargestellt werden können und deshalb eine Indirektheitsstufe über ein Symbol oder einen Namen erfordern.

Das Konzept des Manipulierens beinhaltet eine unmittelbare Visualisierung der bewirkten Veränderungen. Komplexere Operationen werden in einzelne Schritte von Objektselektion und Funktionsanwendung gegliedert. Dies entspricht dem Interpunktionsstyp 1. Auch hier lassen sich bei realen Systemen Bereiche feststellen, in denen der Typ 2 zur Anwendung kommt (z.B. die Auswahl mehrerer Objekteigenschaften in einem Eigenschaftsfenster).

Diese Einordnung von Systemeigenschaften auf den Grunddimensionen zeigt, daß es keine einzelne Konstellation von Ausprägungen gibt, die die realen Systeme vollständig beschreibt. In dem von den Dimensionen aufgespannten Gestaltungsraum belegen die genannten Systeme mehrere Zellen. Eine einzelne Typenkombination kann deshalb nur einem eingegrenzten Interaktionskontext zugeordnet werden. Es kann aber angegeben werden, welche Ausprägungen in einem System notwendigerweise vorhanden sein müssen, damit von Manipulation gesprochen werden kann. Dies wären im Fall von DM parallele Repräsentation, deiktische Referenz und eine Interpunktion vom Typ 1. Der Zusatz "direkt" wäre dabei eher dem Bereich der Benutzungseigenschaften zuzuordnen, da er nicht objektiv als Systemeigenschaft, sondern nur indirekt durch ergonomische oder psychologische Untersuchungen erfaßt werden kann.

#### Schlußbemerkung

Das dargestellte Modell und die daraus abgeleiteten Dimensionen können zum einen dazu dienen, eine klarere Strukturierung und eine einheitlichere Begriffsverwendung in dem komplexen Gebiet der Software-Ergonomie zu unterstützen. Darüberhinaus sind die dargestellten Kategorien als möglicher Ausgangspunkt für empirische Untersuchungen zu betrachten, bei denen für bestimmte Systemeigenschaften entsprechende Benutzungseigenschaften abgeleitet werden.



Schließlich kann der vorgestellte Ansatz für ein systematischeres Vorgehen im Systemgestaltungsprozeß herangezogen werden. Durch die Dimensionen wird ein Gestaltungsraum möglicher Interaktionsformen aufgespannt. Die einzelnen Zellen dieses Gestaltungsraumes können für ein bestimmtes Gestaltungsproblem in systematischer Weise betrachtet und bewertet werden. In einigen Fällen stößt man dabei auf Kombinationen von Ausprägungen, die bislang nicht realisiert wurden. Auf diese Weise kann der vorgestellte Modellrahmen auch dazu dienen, die Kreativität bei der Entwicklung neuer Interaktionsformen zu fördern.

#### Literaturverzeichnis

- Draper, S. (1986): Display managers as the basis for user-machine communication. In: Norman, D. & Draper, S. (Hrsg.): User Centered System Design, Lawrence Erlbaum, Hillsdale N. J.
- Fähnrich, K. P. & Ziegler, J. E. (1985): Workstations using direct manipulation as interaction mode. In: Shackel, B. (Hrsg.): Human-Computer Interaction - INTERACT '84, North-Holland, Amsterdam-New York-Oxford, pp. 693 - 698.
- Hutchins, E. L.; Hollan, J. D. & Norman, D.A. (1986): Direct manipulation interfaces. In: Norman, D. & Draper, S. (Hrsg.): User Centered System Design, Lawrence Erlbaum, Hillsdale N. J.
- Martin, J. (1973): Design of man-computer dialogues, Prentice-Hall, Englewood Cliffs N. J.
- Moran, T. P. (1981): The command language grammar - a representation for the user interface of interactive computer systems. Int. Journal Man-Machine Studies (15), pp. 3 - 50.
- Nielsen, J. (1986): A virtual protocol model for computer-human interaction. Int. Journal Man-Machine Studies (24), 3.
- Shneiderman, B. (1982): The future of interactive systems and the emergence of direct manipulation. Behaviour and Information Technology, 3, pp. 237 - 256.
- Shneiderman, B. (1987): Designing the user interface: Strategies for effective human-computer interaction. Addison-Wesley, Reading MA.
- Ziegler, J. E.; Eichhorn, R.; Hoppe, H. U. & Puetz, R. (1986): Direct manipulation in a general framework of human-computer interaction. Tech. Rep. ESPRIT Project 385 (HUFIT), Fraunhofer-Inst. IAO, Stuttgart.

Jürgen E. Ziegler  
 Fraunhofer-Institut IAO  
 Holzgartenstr. 17  
 7000 Stuttgart 1



## **DISKUSSIONSGRUPPEN**