

Normen der Dialoggestaltung

Quelle: „Gebrauchstauglichkeit von Benutzerschnittstellen“ Susanne Annen

Ziel des Dialoges ist eine Benutzerführung. Der Anwender soll bei der Erledigung seiner Aufgabe unterstützt werden. Ein guter Dialog vermeidet dabei jegliche Über- oder Unterforderung des Menschen. Die vorher durchgeführte **Benutzeranalyse** hilft bei der Umsetzung dieser Forderung. Ein Dialog sollte selbsterklärend sein, d.h., dass der Anwender den Dialog selbstständig, ohne fremde Hilfe benutzen kann. Das bedeutet jedoch nicht, dass auf die Einführung eines Anwendungssystems, die mit einer Schulung verbunden sein kann, verzichtet werden darf. Weiterhin ist durch die Dialoggestaltung sicherzustellen, dass das System vom Benutzer akzeptiert wird. Die Anpassung des Dialoges an die Fähigkeiten des Benutzers sollte ebenso möglich sein. Diese kann entweder durch den Benutzer erfolgen, oder durch das System automatisch vorgenommen werden. Um einen Dialog nach seiner Brauchbarkeit zu bewerten, werden angemessene Kriterien benötigt. Benutzungsschnittstellen sind jedoch sehr inhomogene und vieldimensionale Gebilde, die sich weitgehend einer direkten, objektiven Messung entziehen. Dennoch gibt es eine Reihe von Merkmalen, die eine Bewertung ermöglichen. Die ISO 9241-10 definiert sieben Grundsätze der Dialoggestaltung. Diese sind nicht unabhängig voneinander. So kann es nötig sein, Vorteile des einen Grundsatzes gegenüber denen eines anderen abzuwägen. Im Folgenden werden die Grundsätze einzeln beschrieben.

Aufgabenangemessenheit

Die ISO 9241-10 definiert einen Dialog als "aufgabenangemessen, wenn er den Benutzer unterstützt, seine Arbeitsaufgabe effektiv und effizient zu erledigen."

Ein Dialog sollte nur die Informationen anzeigen, die für die zu erledigende Aufgabe relevant sind. Damit soll eine Reizüberflutung des Benutzers vermieden werden. Entwickler bieten oft überflüssige Informationen an, weil beispielsweise die bei der Programmierung verwendete Komponente diese Funktionalität bietet. Die Frage, ob damit die Erledigung der Arbeitsaufgabe erleichtert wird, wird oft nicht gestellt. Die Darstellung der Informationen muss dabei an die Arbeitsaufgabe angepasst werden. So sollten beispielsweise lange Listen nach Vorgabe des Benutzers sortiert werden können. Die Anzeige der Genauigkeit von Zahlen wird nicht von den technischen Möglichkeiten, sondern von der Aufgabe vorgegeben. Aufgabenangemessenheit bedeutet auch, dass das Dialogsystem automatisierbare Arbeitsschritte auch selbstständig ausführt. So können Positionsmarken im vornherein sinnvoll platziert oder Startprozeduren automatisch vom System ausgeführt werden. Bei der Dialoggestaltung ist sowohl der Grad an Komplexität der Aufgabe als auch die Qualifikation des Benutzers zu beachten. So müssen beispielsweise bei komplizierten Aufgaben die Hilfetexte die einzelnen Teilprobleme entsprechend umfassend erläutern. Die auf Verlangen des Benutzers angezeigte Hilfe sollte sich mit der gegenwärtig zu bearbeitenden Aufgabe beschäftigen. Innerhalb des Hilfesystems kann dann der Benutzer zu anderen Themenbereichen navigieren. Eine weitere Forderung besteht in der Vorgabe von Standardwerten. Diese müssen sinnvoll ausgewählt und durch den Benutzer veränderbar sein. Vor allem bei Datumseingaben tritt häufig der Fall auf, dass das aktuelle Datum eingegeben werden muss. Das Eingabefeld kann in diesem Fall vom Anwendungssystem gefüllt werden. Enorme Zeiteinsparungen lassen sich auch erreichen, wenn häufig wiederkehrende Aufgaben speziell behandelt werden können. Es sollte möglich sein, diese zu automatisieren und damit zu beschleunigen. Im Allgemeinen wird das durch den Einsatz von Makros realisiert. Werden während der Erledigung einer Aufgabe Daten geändert, sollte dem Benutzer möglich sein, die ursprünglichen Daten wieder herzustellen. So kann beispielsweise die Eingabe in einem Textfeld durch Drücken der Taste ESC wieder zurückgesetzt werden. Deutlich mächtiger als dieser einfache Mechanismus ist eine Undo-Funktion, die ganze Dialogschritte

rückgängig machen kann. Die letzte Forderung besteht in der Vermeidung unnötiger Arbeitsschritte. So kann beispielsweise das Drucken und Schließen einer Rechnung in einem Schritt durchgeführt werden.

Die Effizienz des Dialoges hängt entscheidend von der Anpassung des Systems an die Aufgabe ab. Die benutzergerechte Gestaltung kann mit der Anpassung an die Aufgabe konkurrieren. In diesen Fällen muss ein Kompromiss gefunden werden.

Selbstbeschreibungsfähigkeit

"Ein Dialog ist selbstbeschreibungsfähig, wenn jeder einzelne Dialogschritt durch Rückmeldung des Dialogsystems unmittelbar verständlich ist oder dem Benutzer auf Anfrage erklärt wird."

Diese Erläuterungen stellen eine Ergänzung der vorher durchgeführten Benutzerschulung dar, keinen Ersatz. Hilfetexte sollen es dem Benutzer erleichtern, sich schnell und problemlos ein mentales Modell des Anwendungssystems aufzubauen. Die Erläuterungen müssen den Kenntnissen des Benutzers angepasst sein. So kann vor allem bei Anfängern die Darstellung von Beispielen hilfreich sein. Die Texte sollten sowohl die Muttersprache des Anwenders als auch die Fachbegriffe des Arbeitsgebietes verwenden. Ein Dialogsystem sollte nach jeder Handlung des Benutzers eine Rückmeldung geben. Wird vom Benutzer ein Arbeitsschritt initiiert, der schwerwiegende Folgen haben kann, sollte zuvor eine Bestätigung verlangt werden. Dabei sollte der Benutzer die Möglichkeit haben, sich dazugehörige Erläuterungen anzeigen zu lassen. Das Dialogsystem sollte sicherstellen, dass der Benutzer immer über den gegenwärtigen Dialogzustand informiert ist. Dazu gehört beispielsweise ein Überblick über vergangene und zukünftige Dialogschritte. Wird eine Eingabe verlangt, sollten dem Benutzer Informationen darüber angezeigt werden. Ein Beispiel ist die Anzeige des erwarteten Formates. Das Hilfesystem sollte die Erläuterungen kontextabhängig anbieten. Das bedeutet, dass in Abhängigkeit der aktuellen Dialogsituation ein an den Anwendungszustand orientierter Text angeboten wird. Merkmale der Dialogsituation stellen beispielsweise das ausgewählte Fenster, der Eingabefokus, oder der aktuelle Bearbeitungsschritt der Aufgabe dar. Hilfetexte können aktive oder passiv sein. Ersteres bedeutet, dass der Benutzer die Initiative übernimmt und durch eine entsprechende Eingabe die Anzeige von Erläuterungen fordert. Aktive Hilfe sind automatisch vom System angebotene Erläuterungen. Ein Beispiel wäre die Anzeige eines Hilfetextes nach einem Eingabefehler.

Steuerbarkeit

"Ein Dialog ist steuerbar, wenn der Benutzer in der Lage ist, den Dialogablauf zu starten sowie seine Richtung und Geschwindigkeit zu beeinflussen, bis das Ziel erreicht ist."

Das heißt, dass die Kontrolle, soweit es die Aufgabe zulässt, beim Benutzer liegen soll. Parallel durchführbare Aufgaben müssen auch gleichzeitig abzuarbeiten sein. Das System sollte in mehrere unterbrechbare und wieder fortsetzbare Teildialoge gegliedert werden. Auf modale Dialoge, die einen Wechsel in einen anderen Dialog ohne vorherige Beendigung verhindern, sollte beim Systementwurf verzichtet werden. Ausgeschlossen sind kritische Vorgänge, wie zum Beispiel Fehlermeldungen. Bei Systemen mit hohem Sicherheitsbedürfnis ist ebenfalls von zu großer Freiheit für den Benutzer abzu-sehen und ein systemgesteuerter Dialog vorzuziehen. Die Geschwindigkeit des Dialoges sollte dem Benutzer nicht vorgeschrieben werden. So wird beispielsweise das Ende der Eingabe in einem Feld mit der Taste ENTER durch den Benutzer signalisiert. Der Benutzer sollte selbst die Interaktionsform auswählen können. So bevorzugen Anfänger beispielsweise die Funktionsauswahl über Menüs, während Experten lieber Kommandos verwenden. Die ISO 9241 subsumiert unter dem Begriff Steuerbarkeit

auch die Forderung nach Stornierbarkeit von Aktionen. Es sollte wenigstens der letzte Dialogschritt, soweit seine Folgen reversibel sind, zurückgenommen werden können. Diese Technik wird als Undo bezeichnet. Durch die Möglichkeit, die letzte Aktion rückgängig zu machen, wird der Anwender ange-regt, das Anwendungssystem zu explorieren. Die Angst, einen Fehler zu begehen wird entscheidend gesenkt. Soweit möglich sollte das System nicht nur für den letzten, sondern für alle gemachten Schritte ein Undo ermöglichen. Möchte der Benutzer etwas ausprobieren, ist meist eine gewisse Anzahl von Arbeitsschritten notwendig, um das Ergebnis zu erhalten. Wird diese Technik unterstützt, sollte die direkte Wahl des entsprechenden Undo-Schrittes möglich sein, um mehrere Arbeitsschritte auf einmal rückgängig zu machen. Da sich der Benutzer nur bis zu einem gewissen Grad die letzten Aktionen merken kann, muss in diesen Fällen die Anzeige der Schritte auch eine kurze Erläuterung beinhalten. Die Rücknahme einer Undo-Operation wird mit Redo bezeichnet.

Erwartungskonformität

"Ein Dialog ist erwartungskonform, wenn er konsistent ist und den Merkmalen des Benutzers entspricht, z. B. seinen Kenntnissen aus dem Arbeitsgebiet, seiner Aus-bildung und seiner Erfahrung sowie den allgemein anerkannten Konventionen."

Während sich die Erfahrungen des Benutzers bei der Arbeit mit dem System immer mehr ausweiten, schränkt sich die Menge der Erwartungen immer mehr ein. Stimmen die Erwartungen des Benutzers mit dem tatsächlichen Systemverhalten überein, so ist das Anwendungssystem konsistent. Dazu sollte Dialogverhalten und Informationsdarstellung innerhalb des Dialogsystems einheitlich sein. Hierbei helfen sowohl der Einsatz von Standardkomponenten als auch Style Guides. Eine Änderung des Dialogzu-standes sollte auf einheitliche Weise herbeigeführt werden. Dazu zählen der einheitliche Aufruf des Hilfesystems oder der Abbruch eines Dialogs mit der Taste ESC. Der Einsatz eines einheitlichen, dem Benutzer vertrauten Wortschatzes steigert nicht nur die Selbstbeschreibungsfähigkeit sondern auch die Erwartungskonformität. Die Gestaltung von ähnlichen Aufgaben sollte immer gleich sein. Damit kann der Benutzer ein bekanntes Verfahren leicht abgewandelt wieder einsetzen. Zusätzlich ist ein hoher Grad an Konsistenz von Bedeutung, da er das Lernen vereinfacht und unnötige Belastungen für den Benutzer vermeidet. Es lassen sich zwei Arten von Konsistenz unterscheiden. Die Einheitlichkeit der Dialoge innerhalb einer Anwendung wird als innere Konsistenz bezeichnet. Sie muss durch den Dialogentwickler sichergestellt werden. Dabei haben Verwendung, Positionierung, Farbe und Verhal-ten von Bildelementen einen großen Einfluss. Die Äußere Konsistenz beschreibt die Einheitlich-keit von Dialogen verschiedener Anwendungssysteme. In diesem Bereich wirken die Richtlinien und Techniken des zugrunde liegenden Betriebssystems unterstützend.

Fehlertoleranz

"Ein Dialog ist fehlertolerant, wenn das beabsichtigte Arbeitsergebnis trotz er-kennbar fehlerhafter Eingaben entweder mit keinem oder mit minimalem Korrek-turaufwand seitens des Benutzers erreicht werden kann."

Fehler und Probleme im Umgang mit Software treten alltäglich auf. Etwa 10 Prozent der Computerar-beitszeit verbringen Benutzer damit, diese Fehler zu bewältigen. Damit stellen sie nicht nur eine zu-sätzliche Belastung für den Benutzer dar, sondern verursachen ferner für das Unternehmen hohe Kos-ten. Fehler sind immer vom aktuellen Arbeitskontext abhängig und bedürfen daher spezifischer Be-handlung. Syntaktische Fehler sind vom System erkennbar. Eine Korrektur kann automatisch oder ma-nuell erfolgen. Wird der Fehler automatisch korrigiert, beispielsweise Tippfehler, muss der Benutzer darauf hingewiesen werden, ohne jedoch den Dialogablauf zu stören. In dem genannten Fall kann die Eingabe korrigiert angezeigt werden, was als Hinweis ausreichend ist. Die automatische Fehlerkorrek-tur muss abschaltbar sein. Bei manueller Fehlerbeseitigung muss eine Fehlermeldung erfolgen, die in

ihrer Ausführlichkeit anpassbar ist. Das erlaubt unerfahrenen Benutzern sich über den Grund des Fehlers ausführlich zu informieren, während Experten mit diesen Informationen nicht belästigt werden. Zusätzlich sollte der Benutzer bei der Behebung des Fehlers unterstützt werden. Vorhandene Alternativen können dargestellt und erläutert werden. Die Fehlermeldungen selbst sind verständlich, sachlich und konstruktiv zu formulieren. Sie sollten einheitlich strukturiert werden. Dazu gehört sowohl der Aufbau des Fehlertextes als auch seine Darstellung.

Das Ziel beim Entwurf des Anwendungssystems besteht darin, das Auftreten von Fehlern zu vermeiden. So können Eingaben beschränkt werden, indem z.B. keine Ziffern in Textfeldern mit Zahlen als Inhalt erlaubt sind.

Semantische Fehler treten immer dann auf, wenn das mentale Modell des Benutzers von dem tatsächlichen Modell des Anwendungssystems abweicht. Sie können vor allem durch erwartungskonforme Systeme und den Einsatz geeigneter Metaphern eingegrenzt werden.

Individualisierbarkeit

"Ein Dialog ist individualisierbar, wenn das Dialogsystem Anpassungen an die Erfordernisse der Arbeitsaufgabe sowie an die individuellen Fähigkeiten und Vorlieben des Benutzers zulässt."

Umfassende Individualisierungsmöglichkeiten sind keine Rechtfertigung, auf die ergonomische Gestaltung von Dialogen zu verzichten. Die Möglichkeiten der Anpassung müssen durch den Entwickler sinnvoll begrenzt werden. Damit sollen Beeinträchtigungen durch Individualisierung vermieden werden. Beispielsweise darf es nicht möglich sein, die Lautstärke eines Audiosystems soweit zu erhöhen, dass Hörschäden entstehen können.

Grundsätzlich sollten Techniken zur Anpassung an Sprache und kulturelle Eigenheiten bereitgestellt werden. Dieser Aspekt wird unter dem Begriff Internationalisierung diskutiert. Weiterhin sollten sensorische und geistige Fähigkeiten des Benutzers beachtet werden. So ist beispielsweise für farbenfehlsichtige Anwender die Anpassung von Farben ein wichtiges Kriterium. Das Wissen der Benutzer und ihre Erfahrungen im Aufgabenbereich und mit dem Anwendungssystem sind unterschiedlich und unterliegen einem stetigen Wandel. Dementsprechend sollten Individualisierungsmöglichkeiten auch diesen Bereich abdecken. Ein Beispiel ist die Anpassung des Umfangs von Hilfetexten. Die Darstellung von Informationen sollte ebenfalls änderbar sein. So bevorzugen manche Benutzer eine Listendarstellung, während andere die Darstellung in Karteiform präferieren. Auch wenn in der Anwendung der Wortschatz des Arbeitsgebietes verwendet wurde, sollte es dem Benutzer erlaubt sein, eigene Bezeichnungen festzulegen. Die letzten Forderungen sind Anpassung von Kommandos und die Wahl der Interaktionsform. Die Kommandoanpassung umfasst sowohl ihre Bezeichnung als auch die Funktionalität.

Lernförderlichkeit

"Ein Dialog ist lernförderlich, wenn er den Benutzer beim Erlernen des Dialogsystems unterstützt und anleitet."

Lernen als das Erarbeiten einer möglichst geeigneten Vorgehensweise oder Strategie zur Erfüllung einer gegebenen Aufgabe durch wiederholte Übung ist für die effiziente Nutzung eines Softwaresystems von großer Bedeutung. Umfangreiche Systeme können auch durch einführende Schulungen nicht von Anfang an umfassend erfasst und in ihrer ganzen Funktionalität genutzt werden. Vielmehr sollte der Benutzer unterstützt werden, selbstständig sein Wissen über das System während seiner Nutzung zu

verbessern und zu erweitern. Dabei helfen die bereits erwähnten Metaphern, die es ermöglichen, bereits vorhandenes Wissen in den neuen Bereich zu transferieren. Eine Hilfefunktion wirkt ebenfalls unterstützend. Die bereits erläuterte Undo-Funktion unterstützt das "Learning-by-doing". Bei diesem Konzept soll der Benutzer während der Erledigung einer Aufgabe lernen. Werden Schritte ausgeführt, die fehlerhafte Auswirkungen haben, können sie jederzeit wieder rückgängig gemacht werden.

Die Lernförderlichkeit eines Anwendungssystems ist auch wichtig, weil Benutzer die Eigenschaft besitzen, Funktionen und Abläufe, die länger nicht genutzt wurden, wieder zu vergessen. In diesen Fällen reicht ein kurzes Wiederauffrischen des bereits Gelernten. Kurze Hinweise in Form von Tipps können dabei bereits ausreichen. Ein wichtiger Nebeneffekt besteht darin, dass Lernen bei der Benutzerzufriedenheit einen positiven Einfluss besitzt.