

Ergonomische Aspekte der Software-Gestaltung

Teil 6: Evaluation von Software

von Gert Zülch, Axel E. Fischer und Sascha Stowasser

Schwerpunkt der Software-Ergonomie, die als spezielles Teilgebiet der Arbeitswissenschaft gilt, ist die Analyse, Gestaltung und Bewertung von rechnerunterstützten, interaktiven Informations- und Kommunikationssystemen. In den bisherigen Teilen dieser Serie wurden verschiedene Ebenen der Benutzungsschnittstelle zwischen Mensch und Rechner definiert, um den Informationsaustausch zwischen Mensch und Rechner zu analysieren und zu gestalten. Dabei wurde auf der untersten Ebene die Informationscodierung (Teil 3 dieser Serie in Ausgabe 1/1998) und darauf aufbauend die Informationsanordnung auf dem Bildschirm (Teil 4 in Ausgabe 2/1998) erörtert. Anschließend wurde die Gestaltung des dynamischen Informationsaustausches zwischen Benutzer und Rechner in das Ebenenmodell der Benutzungsschnittstelle integriert (Teil 5 in Ausgabe 3/1998). Dabei wurde darauf hingewiesen, daß sich die Dialoggestaltung an den Eigenschaften und Fähigkeiten des Benutzers zu orientieren hat.

Eine wesentliche Aufgabe der Software-Ergonomie besteht darin, die Anforderungen, die der Benutzer an ein Programmsystem stellt, in geeigneter Weise bei dessen Herstellung zu berücksichtigen. In Anlehnung an ENGLISCH (1992) beschreibt die Benutzungsfreundlichkeit, wie gut Programmsysteme an die zu bearbeitenden Aufgaben sowie an die Fähigkeiten und Bedürfnisse der Benutzer angepaßt sind. Damit ist der wesentliche Gegenstand einer benutzungsfreundlichen Gestaltung das Programmsystem selbst (vgl. ZÜLCH, KIPARSKI, GRIEBER 1997). In diesem Beitrag wird auf die Notwendigkeit zur Evaluation und die Vorgehensweise bei der Evaluation von Software eingegangen. Abschließend wird eine Evaluationsmethode vorgestellt, die bereits praktisch erprobt ist.

Notwendigkeit der Software-Evaluation

Analysen zur Benutzungsfreundlichkeit von Software sollen bei der Konzipierung, der Auswahl, dem Erwerb und bei wesentlichen Änderungen vorgenommen werden (vgl. ENGLISCH 1992), um dem Entwickler ein benutzerorientiertes Feedback bei der Erstellung der Software zu geben. An einer Beurteilung der Benutzungsfreundlichkeit sollten daher neben den Benutzern auch der Auftraggeber (d.h. der Käufer oder firmeninterne Besteller), der Software-Hersteller sowie ggf. auch Softwareprüfstellen ein verstärktes Interesse haben (vgl. BERGFELD-MILLS 1986; WIEKEN

1990; ENGLISCH 1992; SMITH, WILLIAMS 1997). Somit sind entsprechende Evaluationen von der Entwicklung bis zur Nutzung der Software aus unterschiedlichen Anlässen notwendig (vgl. ZÜLCH, ENGLISCH, GRUNDEL 1993).

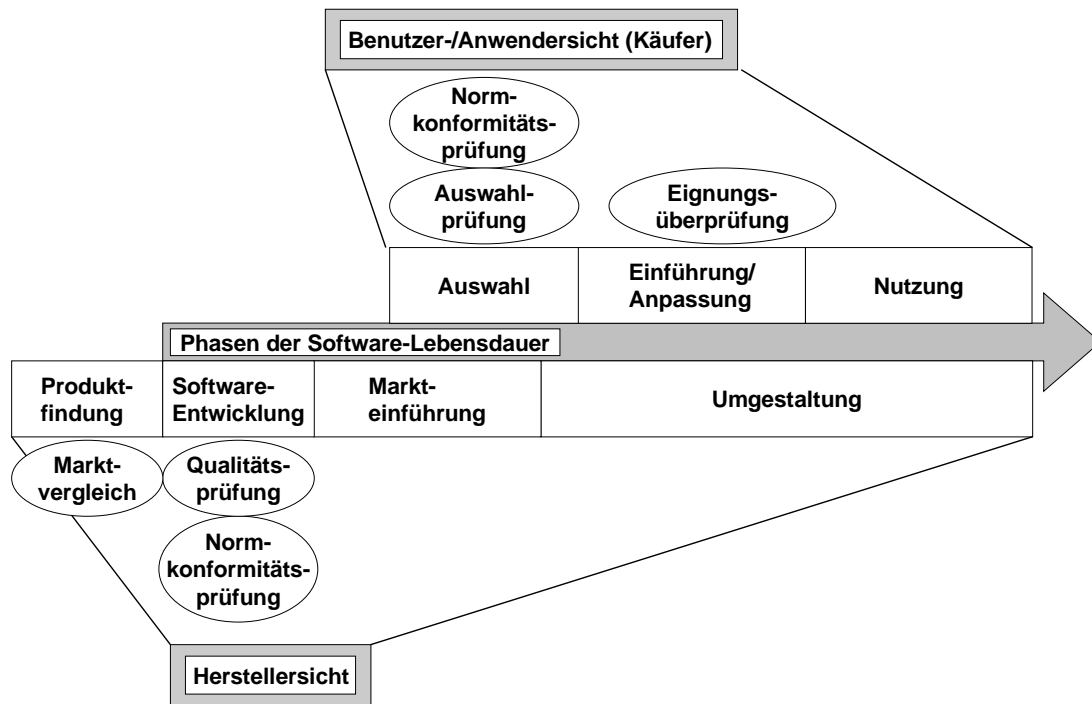


Bild 1: Anlässe zur Evaluation von Programmsystemen

Zum einen kann die Evaluation von Programmsystemen durch den Softwarehersteller, zum anderen durch den Anwender, also der Käufer eines Programmsystems, veranlaßt oder auch selbst vorgenommen werden. Für den Hersteller eines Programmsystems ist es interessant, ob die erstellte Software die geforderten Anforderungen unter funktionalen und benutzerorientierten Aspekten erfüllt. Mit Hilfe einer Evaluation kann bereits während des Entwicklungsprozesses eine diesbezügliche Qualitätsprüfung durchgeführt werden. Bei der Qualitätsprüfung wird beurteilt, wie gut sich ein Programm für ein Anwendungsgebiet eignet und ob spezielle Anforderungen der Benutzungssituation erfüllt werden. Zudem kann gegen Ende des Entwicklungsprozesses die Normkonformität der Software dahingehend überprüft werden, ob bestimmte Mindestanforderungen erfüllt werden.

Für den Anwender stellt sich bei der Auswahl eines Programmsystems, das eine bestimmte Arbeitsaufgabe erfüllen soll, die Frage, wieviele Systeme verschiedener Hersteller zu welchen Preisen auf dem Markt verfügbar sind. Daneben sollte auch von Bedeutung sein, wie die Benutzungsoberfläche und die Interaktion zwischen Mensch und Maschine gestaltet ist. Beim Marktvergleich werden unterschiedliche, auf dem Markt befindliche Programmsysteme miteinander verglichen, um herauszufinden, welches Programm am besten zur Erfüllung der Arbeitsaufgabe

geeignet ist. Zudem kann ein Anwender, wie auch der Hersteller, Normkonformitätsprüfungen durchführen. Bei der Normkonformitätsprüfung werden Programmsysteme auf die Einhaltung der in einer Norm gestellten Anforderungen untersucht. Als Ergebnis kann festgehalten werden, zu wieviel Prozent das entsprechende Programmsystem die Vorgaben einer Norm erfüllt (ENGLISCH 1992; ZÜLCH, ENGLISCH, GRUNDEL 1993), also wie normkonform es ist. Der Benutzer hingegen wählt vor jeder Erfüllung einer Arbeitsaufgabe ein vom Anwender zur Verfügung gestelltes Programmsystem aus, mit dessen Hilfe er die Aufgabe bearbeitet.

Benutzungsfreundlichkeit

Damit die Benutzungsfreundlichkeit von Programmsystemen möglichst objektiv beurteilt werden kann, muß zunächst der Begriff "Benutzungsfreundlichkeit" definiert werden. Die Beurteilung der Benutzungsfreundlichkeit gestaltet sich als schwierig, da der Begriff in der Literatur unterschiedlich definiert wird.

Faßt man die Grundsätze der Dialoggestaltung (z.B. Aufgabenangemessenheit, Erwartungskonformität, Selbstbeschreibungsfähigkeit, Steuerbarkeit, Fehlerrobustheit; vgl. DIN/ISO 9241, Teil 10) als Definition der Benutzungsfreundlichkeit auf, läßt sich diese an konkreten Eigenschaften festmachen. Nach DZIDA, HERDA und ITZFELDT (1987) ist die Benutzungsfreundlichkeit das umfassende Bewertungsmerkmal, dem die verschiedenen Einzelmerkmale als Bewertungsaspekte zugeordnet sind. Somit sind die in der DIN/ISO 9241 aufgeführten Empfehlungen als Evaluationsmerkmale verwendbar. Die DIN/ISO 9241 gliedert sich in 17 Teile, die auch Hardware-Aspekte mit abdecken. (siehe Bild 2).



Bild 2: Aufbau der DIN/ISO 9241

Als besonders unangemessen wird ein Programmsystem bezüglich seiner Benutzungsfreundlichkeit dann angesehen, wenn die Benutzungsfreundlichkeit als "freundlich-geschwätziges Verhalten des Systems gegenüber dem Benutzer verstanden wird" (WANDMACHER 1993, S. 200). Dieses Systemverhalten drückt sich z.B. in pseudopersonlichen Meldungen zur Begrüßung des Benutzers aus, wodurch Gefühls- und Beziehungsfähigkeiten des Systems nahegelegt werden sollen.

Vorgehensweise bei der Evaluation

Die Vorgehensweise bei der Evaluation eines Programmsystems läßt sich durch das Basismodell einer Evaluation (Bild 3) beschrieben. Zur Evaluation eines beliebigen Produktes können grundsätzlich verschiedene Tools (Werkzeuge) verwendet werden. Grundlage dieser Tools sind Evaluationsmethoden und -verfahren, die auf allgemeinen Anforderungen an das Produkt basieren. Diese Anforderungen können aus den Bedürfnissen der Benutzer, den durch das Programmsystem zu erfüllenden Aufgaben und aus dem Stand der Technik abgeleitet werden. Die Anforderungen (z.B. ergonomischer Art) werden in Kriterien oder Leitlinien erfaßt, die das Produkt erfüllen muß. Bei der Evaluation wird - möglichst unter Zuhilfenahme eines Evaluationstools - überprüft, inwieweit das Produkt den gestellten Anforderungen gerecht wird, also die Kriterien erfüllt oder den Leitlinien entspricht.

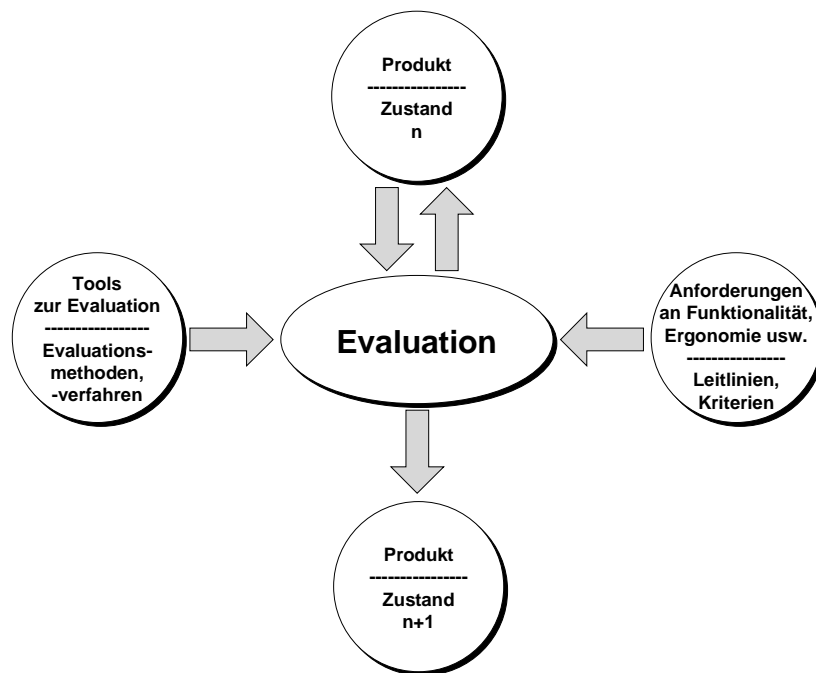


Bild 3: Basismodell einer Evaluation

Die an das Produkt bzw. das Programmsystem zu stellenden ergonomischen Anforderungen (siehe Grundsätze der Dialoggestaltung) sind in den bisherigen Teilen dieser Serie bereits erläutert worden (vgl. ZÜLCH, STOWASSER, FISCHER 1998). Nachfolgend soll auf verschiedene Methoden eingegangen werden, die zur Evaluation angewandt werden können. Die wichtigsten Methodengruppen sind (vgl. ENGLISCH 1992; GRIEßER 1995):

- Befragungen,
- theoriebasierte Evaluationsmethoden,
- Leitfäden und
- experimentelle Untersuchungen.

Befragungen

Befragungen gehören zu den klassischen Standarderhebungstechniken der sozialwissenschaftlichen Forschung. Grundsätzlich unterscheidet man schriftliche (Fragebogen) und mündliche (Interviews) Befragungen. In beiden Fällen wird die Meinung des Benutzers über bestimmte Eigenschaften eines Programmsystems festgestellt, die dieser dann aufgrund seiner bisher mit diesem System gemachten Erfahrungen beantworten muß. In Bild 4 ist beispielhaft eine Checkliste dargestellt, die sowohl als Fragebogen als auch als Interviewleitfaden eingesetzt werden kann. Dabei steht in der zweiten Spalte das "F" für eine Frage, die an den Bildschirmgeräte-Benutzer gestellt wird, während

das "P" eine Prüfung des Sachverhaltes bedeutet. Die Abkürzung "MW" in Spalte sechs steht für Mittelwert (ZÜLCH, KIPARKI, GRIEBER 1997).

Nr.		Überprüfung der Software	Soll	i.O.	Bemerkungen
S 01	F	Sind Sie im Hinblick auf ihre Tätigkeit mit dem Programm zufrieden? (Bedienung, logischer Aufbau)	ja		
S 02	F	Gibt das Programm Informationen über den jeweiligen Bearbeitungszustand?	ja		
S 03	F	Sind Sie mit der Darstellung der Systeminformation zufrieden? (Umfang, Form, Geschwindigkeit)	ja		
S 04	P	Ist die Zeichendarstellung positiv?	ja		
S 05	P	Sind die Zeichen scharf und deutlich erkennbar?	ja		
S 06	P	Sind die Zeichen eindeutig? (i, I, o, O, ü, Ü)	ja		
S 07	P	Können alle notwendigen Zeichen dargestellt werden?	ja		
S 08	P	Ist die Zeichengröße ausreichend?	ja		MW: DIN ISO 9241 Teil 3 verlangt eine Schrifthöhe von $\geq 2,6$ mm bei einem Bildschirmabstand von ≥ 400 mm
S 09	P	Ist der Zeilenabstand ausreichend?	ja		

Bild 4: Auszug aus einem Erfassungsbogen zur Überprüfung von Software (Quelle: ZÜLCH, KIPARSKI, GRIEBER 1997)

Dabei kann der Benutzer seine Erfahrungen entweder bei der Benutzung der Software gewonnen haben, oder der Untersuchungsleiter definiert eine bestimmte Aufgabe, anhand derer der Benutzer speziell für die Durchführung der Befragung seine Meinung äußern kann. Der Vorteil liegt im ersten Fall in der größeren Erfahrung, die der Benutzer des Systems ggf. über längere Zeit gesammelt hat. Im zweiten Fall hingegen sind die Ergebnisse eher vergleichbar, da die Benutzer eine definierte Mindesterfahrungsbasis mit dem System besitzen.

In beiden Fällen werden vorrangig subjektive Daten erhoben, vor allem dahingehend, ob die Benutzung eines Systems angenehm, einsichtig oder klar ist. Unterschieden wird zwischen mündlichen und schriftlichen Befragungen sowie zwischen solchen, die ein Untersuchungsleiter selbst als Benutzer (Durchführungsprotokoll) oder mit einer anderen Person als Benutzer durchführt. Befragungen sind nicht sehr aufwendig und können in allen Phasen einer Systementwicklung angewendet werden. Als problematisch kann sich die Vielzahl der produzierten

Daten und die damit verbundene aufwendige Auswertung und die Neigung zu Übertreibungen der Befragten erweisen sowie die Tatsache, daß Befragungen von den Befragten nicht sonderlich geschätzt werden (OPPERMANN, MURCHNER, REITERER u.a. 1992).

Bei einer Befragung sollte man zwei Grundregeln beachten (OPPERMANN, MURCHNER, REITERER u.a. 1992):

1. Vor der Formulierung von Fragen und Antwortvorgaben sollte man genau überlegen, was die möglichen Antworten für das Experiment und das Untersuchungsanliegen bedeuten. Ansonsten werden durch die Fragen und Antworten Daten produziert, die wenig aussagekräftig sind.
2. Vor dem Einsatz eines vorbereiteten Fragebogens sollte dieser an einigen Repräsentanten der Zielgruppe erprobt werden.

Eine weitere Form der Befragung, allerdings eine Sonderform, stellt das Protokoll des "lauten Denkens" dar. Dabei formuliert der Benutzer während der Erfüllung der Arbeitsaufgabe laut seine Überlegungen, Probleme, Handlungsalternativen usw. und bietet dadurch Anhaltspunkte für die Interpretation des Versuchsablaufs (GRIEBER 1995). Bei der Methode der "konstruktiven Interaktion" bearbeiten zwei Benutzer gemeinsam eine Arbeitsaufgabe (GÖRNER, ILG 1993). Das aufgezeichnete Gespräch liefert hierbei das Auswertungsmaterial.

Eine weitere Möglichkeit besteht darin, dem Benutzer im Anschluß an die Aufgabenbearbeitung, die auf Video aufgezeichnet wurde, diese Videoaufzeichnung vorzuführen. Der Benutzer hat dabei die Aufgabe, seine auf Video festgehaltenen Aktionen zu erläutern und zu interpretieren (Videokonfrontation) (OPPERMANN, MURCHNER, REITERER u.a. 1992).

Theoriebasierte Evaluationsmethoden

Anhand von Modellen der Mensch-Computer-Interaktion werden bei den theoriebasierten Evaluationsmethoden bestimmte Kenngrößen eines Programmsystems, wie beispielsweise die Anzahl der Interaktionen zur Bewältigung einer bestimmten Arbeitsaufgabe, theoretisch abgeleitet. In der Regel sind diese Analysen sehr aufwendig und spielen daher nur in Grundlagenuntersuchungen eine wesentliche Rolle (GRIEBER 1995).

Wichtige Verfahren sind das GOMS-Modell (WANDMACHER 1997), das Keystroke-Level-Modell (WANDMACHER 1993), die Cognitive Complexity Theory (KIERAS, POLSON 1985), die Task Action Grammar (PAYNE 1985), Zustands-Übertragungsnetzwerke (JOHANNSEN 1993) und Cognitive Walkthroughs (POLSON, LEWIS, RIEMAN u.a. 1992).

Leitfäden

Beim Einsatz von Evaluationsleitfäden wird ein Programmsystem von einem oder mehreren Experten anhand einer Prüfliste in Bezug auf seine Benutzungsfreundlichkeit überprüft (RAUTERBERG, SPINAS, STROHM u.a. 1994). Derartige Prüflisten orientieren sich dabei in der Regel weniger an den Aufgabenstellungen, die mit dem Verfahren zu erledigen sind, sondern vielmehr an softwareergonomischen Fragestellungen (OPPERMANN, MURCHNER, REITERER u.a. 1992; GRIEBER 1995).

Die bestehenden Prüflisten variieren sowohl im Grad ihrer Detaillierung als auch im Grad ihrer Einbettung in eine Testsituation (Verfahrensvorschrift für die Durchführung einer Evaluation) sehr stark (BAITSCH, KATZ, SPINAS u.a. 1991; TBS 1993). Während beispielsweise das EVADIS-II-Verfahren neben den Prüffragen eine Durchführungsvorschrift besitzt, sind die als Prüfliste verwendbaren Normen wie z.B. DIN/ISO 9241, Teil 10, (vgl. Bild 5) nicht in eine Testsituation eingebettet (DIN/ISO 9241). Die Normen sind häufig in ihren Empfehlungen zu allgemein gehalten, während die in ihnen gemachten Empfehlungen hingegen zu exemplarisch sind (ZÜLCH, STOWASSER, FISCHER 1998).

Empfehlungen:	mögliche Beispiele:
Regeln und zugrundeliegende Konzepte, die nützlich für das Erlernen sind, sollten dem Benutzer verfügbar sein, damit der Benutzer eigene Ordnungskriterien und Merkregeln für das Einprägen bilden kann.	Der Benutzer kann sich Informationen über das Gedankenmodell, auf dem die Anwendung aufgebaut ist, anzeigen lassen.
Wichtige Lernstrategien (z.B. verständnisorientiertes Lernen, "Learning-by-doing", Lernen am Beispiel) sollten unterstützt werden.	<p>Der Benutzer kann immer zwischen der Übersichts-Hilfeinformation und dem Lernen am Beispiel hin und her schalten (z.B. kann der Benutzer eine Übersicht über die Gesamtstruktur eines Datenbanksystems anfordern. Danach kann der Benutzer eine Erläuterung über eine bestimmte Funktion anfordern und kann diese Funktion in einer "Was wäre wenn?"-Weise ausführen.).</p> <p>"Learning-by-doing" wird dadurch unterstützt, daß der Benutzer ermutigt wird zu experimentieren, in unterschiedlichen Situationen Beispiele durchzuspielen, "Was wäre wenn?"-Alternativen anzuwenden (z.B. Fehlerkorrektur zuzulassen, ohne daß die Gefahr besteht, potentiell katastrophale Ergebnisse herbeizuführen).</p> <p>Interaktiver Online-Unterricht unterstützt das "Learning-by-doing".</p>
Das Erinnern an bereits Gelerntes sollte unterstützt werden.	Auf die Häufigkeit von Kommandos wird geachtet. Häufig benutzte Kommandos bieten dem Benutzer Abkürzungen und Vorbelegungen. Selten benutzte Kommandos sind selbstbeschreibender gestaltet und mit mehr Hinweisse-Informationen versehen.
Eine Reihe unterschiedlicher Mittel zur Verbesserung der Lernförderlichkeit sollte verwendet werden, um dem Benutzer zu helfen, mit den Dialogelementen vertraut zu werden.	<p>Gleichartige Hinweismeldungen erscheinen stets am gleichen Ort im Anzeigenbereich.</p> <p>Für vergleichbare Objekte der Arbeitsaufgabe werden ähnlich gestaltete Bildelemente verwendet.</p>

Bild 5: Empfehlungen zur Lernförderlichkeit aus der DIN/ISO 9241, Teil 10

Experimentelle Untersuchungen

Unter die experimentellen Evaluationsverfahren fällt eine große Anzahl von Ansätzen, ausgehend von einfachen Beobachtungen des Benutzerverhaltens bis hin zu klassischen Experimenten mit einer detaillierten Versuchsplanung (OPPERMANN, MURCHNER, REITERER u.a. 1992). Der große Vorteil der experimentellen Untersuchungsmethoden liegt darin, daß - wie auch bei den

Befragungen - die Benutzer in den Evaluationsprozeß mit einbezogen werden. Die Evaluation kann daher als Feld- oder Laboruntersuchung durchgeführt werden (GÖRNER, ILG 1993). Neben Rechnerprotokoll, Benchmark-Test, Zweitaufgabentechnik, Verhaltensbeobachtung, psychophysiologische Meßmethode und psychologischen Arbeitsanalyseverfahren zählt auch die Blickregistrierung zu den experimentellen Evaluationsverfahren. Dabei wird das Benutzerverhalten bezüglich der visuellen Informationsaufnahme ermittelt (GRIEBER 1995). Der Schwerpunkt der Auswertungen liegt auf der Analyse der Blickpunkte (Fixationen). Diese verhaltensorientierte Methode eignet sich insbesondere zur Analyse von Augenbewegungen während eines Problemlösungsprozesses (RHENIUS, LOCHER 1992). Zusätzlich können Benutzereingaben mittels Keystroke-Recording protokolliert und strukturierte Interviews durchgeführt werden, um die individuellen Vorgehensweisen von Versuchspersonen und deren subjektives Empfinden aufzunehmen.

Evaluation am Beispiel von PROKUS

PROKUS (Programm zur kommunikationsergonomischen Untersuchung rechnerunterstützter Verfahren) ist ein Programmsystem zur Entwicklung und Durchführung von Evaluationsverfahren (ZÜLCH, ENGLISCH, GRUNDEL 1993) und wurde am Institut für Arbeitswissenschaft und Betriebsorganisation (ifab) der Universität Karlsruhe zur Feststellung und Beurteilung der Benutzungsfreundlichkeit von Software entwickelt. Dieses rechnerunterstützte Verfahren kann der Methodengruppe "Leitfäden" zugeordnet werden und unterstützt den Untersuchungsleiter bei der Konzeption und Durchführung einer Evaluation. Die Antworten auf entsprechende Fragen, d. h. die Rohergebnisse der Evaluation, werden in Datenbanken gespeichert und können nach verschiedenen Vorgehensweisen ausgewertet und beurteilt werden. Das Programm kann für Normkonformitätsprüfungen, Marktvergleiche, Auswahlentscheidungen, Eignungsprüfungen und Qualitätsprüfungen verwendet werden. Soll bei einem Programmsystem die Benutzungsfreundlichkeit untersucht werden, sind die bei der Untersuchung zu beantwortenden Fragen zu formulieren. Alle Prüffragen werden in PROKUS in einen festen Bezugsrahmen gesetzt, der aus Angaben zu Komponenten, Kriterien, Methoden und Aufgaben besteht. Daneben können seitens des jeweiligen Anwenders von PROKUS noch drei weitere Bezugfelder (Fragenklassen "Fr.Kl." in Bild 6) frei definiert werden. Diese Bezugfelder werden benötigt, wenn die entsprechenden Prüffrage nur in bestimmten Anwendungsfällen benötigt bzw. relevant wird. Dies ist beispielsweise der Fall, wenn einige Fragen nur bei Textverarbeitungsprogrammen oder bei Spracheingabe verwendet werden sollen. Auf die gestellten Fragen kann seitens des Anwenders nicht frei geantwortet werden, da die Prüffrage aus bestimmten Anforderungen *aus der Norm* (Soll-Vorgaben) resultieren und damit der Sollwert sowie der Antwortbereich festgelegt ist. So kann z.B. die Frage nach der Dauer eines Speichervorganges nur mit der benötigten Zeit beantwortet werden,

wobei die Frage nach dem Vorhandensein einer Hilfefunktion mit ja oder nein beantwortet werden muß.

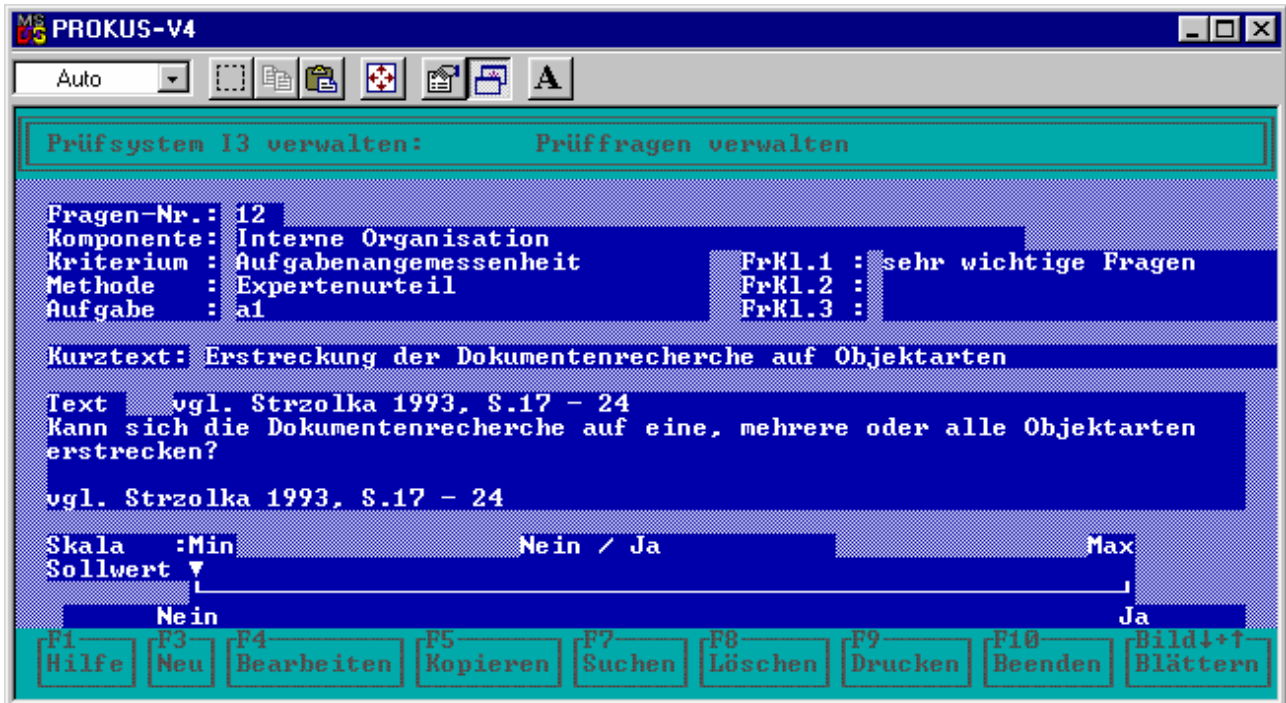


Bild 6: Beispiel einer Prüffrage in PROKUS

Diese Prüffragen können zu Kriterien der Benutzungsfreundlichkeit und zu bestimmten Komponenten des Programmsystems gestellt werden (ZÜLCH, ENGLISCH, GRUNDEL 1993), wie z.B. die Frage "Wird dem Benutzer ein Systemabsturz ohne Zeitverzug (Kriterium "Geschwindigkeit") auf dem Bildschirm (Komponente "Bildschirmanzeige") angezeigt und hierfür geeignete Hilfsmaßnahmen vorgeschlagen?".

Die Verwendung von PROKUS ist vor allem für diejenigen Anwender interessant, die eine große Anzahl von Prüffragen entwickeln, sammeln, pflegen und verwalten wollen, wie z.B. für Software-Prüfstellen oder für Anwender vieler Programmsysteme (z.B. Industrie, Verwaltung usw.).

Ausblick

In der Artikelserie "Ergonomische Aspekte der Software-Gestaltung" wurden bisher die Grundlagen der Software-Ergonomie zusammengefaßt dargestellt. Dabei wurde deutlich, daß die Grundsätze der Software-Ergonomie zwar in verschiedenen nationalen und internationalen Normen festgelegt sind, diese aber sehr allgemein gehalten wurden. Die dadurch entstehende Problematik der Evaluation wurde beschrieben.

Im vorliegenden Beitrag wurden die Möglichkeiten und Anlässe zur Evaluation von Programmsystemen, verwendete Methoden und Vorgehensweisen bei ihrer Durchführung erläutert. Im nächsten Beitrag wird die Evaluation mittels PROKUS exemplarisch dargestellt.

Literatur

BAITSCH, Christof; KATZ, Christian; SPINAS, Philipp u.a.:

Computerunterstützte Büroarbeit.
Zürich: vdf-Verlag, 2. Auflage 1991.

BERGFELD-MILLS, Carol:

Usability testing in the real world.
In: Human Factors in Computing Systems.
Hrsg.: MANTEI, Marilyn; ORBETON, Peter.
New York NY, Boston MA: ACM - Association for Computing Machinery, 1986, S.
212-215.
(CHI'86 Conference Proceedings)

DIN/ISO 9241, Teil 10:

Grundsätze der Dialoggestaltung.
Draft International Standard, August 1993.

DZIDA, Wolfgang; HERDA, Siegfried; ITZFELDT, Wold D.:

Factors of User-Perceived Quality of Interactive Systems.
St. Augustin: Gesellschaft für Mathematik und Datentechnik, 1978.
(Institut für Software-Technologie; Bericht Nr. 40)

ENGLISCH, Joachim:

Ergonomie von Softwareprodukten.
Mannheim u.a.: BI-Wissenschaftsverlag, 1993.
(Angewandte Informatik, Band 5)

ENGLISCH, Joachim:

Systematische Entwicklung von Evaluationsverfahren zur Beurteilung der
Benutzungsfreundlichkeit von CAD-Systemen.
Karlsruhe, Uni Diss. 1992.
(ifab-Forschungsberichte aus dem Institut für Arbeitswissenschaft und Betriebsorganisation
der Universität Karlsruhe, Band 4 - ISSN 0940-0559)

GÖRNER, Claus; ILG, Rolf:

Evaluation der Mensch-Rechner-Schnittstelle.

In: Benutzergerechte Software-Gestaltung.

Hrsg.: ZIEGLER, Jürgen; ILG, Rolf.

München, Wien: R. Oldenbourg Verlag, 1993, S. 189-206.

GRIEBER, Klaus:

Einsatz der Blickregistrierung bei der Analyse rechnerunterstützter Steuerungsaufgaben.

Karlsruhe Uni, Diss. 1995.

(ifab-Forschungsberichte aus dem Institut für Arbeitswissenschaft und Betriebsorganisation der Universität Karlsruhe, Band 10 - ISSN 0940-0559)

HOFFMANN, Thomas; KLOSE, Hans-Georg; MARTIN, Hans:

Handbuch zur software-ergonomischen Gestaltung von Bildschirmmasken.

Düsseldorf: VDI-Verlag, 1989.

(Reihe 10: Informatik/Kommunikationstechnik, Nr. 103)

JOHANNSEN, Gunnar:

Mensch-Maschine-Systeme.

Berlin u.a.: Springer-Verlag, 1993.

KIERAS, David E.; POLSON, Peter G.:

An approach to the formal analysis of user complexity.

In: International Journal of Man-Machine Studies,

London, 27(1985), S. 365-394.

OPPERMANN, Reinhard; MURCHNER, Bernd; REITERER, Harald; KOCH, Manfred:

Softwareergonomische Evaluation.

Berlin, New York: Walter de Gruyter, 1992.

PAYNE, Stephen J.:

Task-Action Grammars.

In: Human-Computer Interaction - INTERACT '84.

Hrsg.: SHACKEL, Brian.

Amsterdam, New York, Oxford: North-Holland, 1985, S. 527-532.

POLSON, Peter G.; LEWIS, Clayton; RIEMAN, John u.a.:

Cognitive walkthroughs: a method for theory-based evaluation of user interfaces.

In: International Journal of Man-Machine Studies,

London, 36(1992), S. 741-773.

RAUTERBERG, Matthias; SPINAS, Philipp; STROHM, Oliver u.a.:

Benutzerorientierte Software-Entwicklung.

Stuttgart: B. G. Teubner Verlag, 1994.

RHENIUS, Detlef; LOCHER, Jürgen:

Auswertungsalgorithmus für Folgen von Augenbewegungen während eines Problemlösungsprozesses.

In: Zeitschrift für experimentelle und angewandte Psychologie,

Göttingen u.a., 39(1992)4, S. 646-661.

SMITH, Connie U.; WILLIAMS, Lloyd G.:

Performance Engineering Evaluation of Object-Oriented Systems with SPE ED.

Hrsg.: MARIE, Raymond u.a.

In: Computer Performance Evaluation, St. Malo, 3.-6. Juni 1997.

Berlin u.a., Springer, 1997, S. 135-154.

TBS - Technologieberatungsstelle beim DGB-Landesbezirk NRW (Hrsg.):

Ergonomieprüfer - Handlungshilfe zur ergonomischen Arbeits- und Technikgestaltung.

Oberhausen: Eigenverlag der TBS, 1993.

(Reihe Technik und Gesellschaft, Heft 14)

WANDMACHER, Jens:

Ein Werkzeug für GOMS-Analysen zur Simulation und Bewertung von Prototypen beim Entwurf.

In: Prototypen für Benutzungsschnittstellen.

Hrsg.: SZWILLUS, G.

Paderborn: Tagungsband PB'97, 1997, S. 35-42.

WANDMACHER, Jens:

Software-Ergonomie.

Berlin, New York: Walter de Gruyter, 1993.

WIEKEN, John-Harry:

Softwareproduktion - Ziele, Vorgehensweisen, Werkzeuge.

Hamburg u.a.: McGraw-Hill, 1990.

ZÜLCH, Gert; ENGLISCH, Joachim; GRUNDEL, Christoph:

Beurteilung der Benutzungsfreundlichkeit von Programmsystemen.

In: Fortschrittliche Betriebsführung und Industrial Engineering,

Darmstadt, 42(1993)5, S. 267-272.

ZÜLCH, Gert; KIPARSKI von, Rainer; GRIEBER, Klaus:
Messen, Beurteilen und Gestalten von Arbeitsbedingungen.
Heidelberg: Dr. Curt Haefner, 1997.

ZÜLCH, Gert; STOWASSER, Sascha; FISCHER, Axel:
Gestaltung des Dialogs zwischen Benutzer und Rechner. Ergonomische Aspekte der
Software-Gestaltung, Teil 5.
In: Ergo-Med,
Heidelberg, 22(1998)3, S.105-109.

Anschrift der Verfasser:

Dipl.-Ing. Axel E. Fischer
Universität Fridericiana (TH) Karlsruhe
Institut für Arbeitswissenschaft und Betriebsorganisation
Kaiserstraße 12
76128 Karlsruhe