

Benutzungszentrierte Schnittstellengestaltung

Ein anpassbares Administrationssystem zur Entwicklung verfahrenstechnischer Prozesse



In diesem Artikel wird die benutzungszentrierte Gestaltung eines anpassbaren Administrationssystems für die Entwicklung verfahrenstechnischer Prozesse vorgestellt. Basierend auf den Aktivitäten, die hauptsächlich mit der Software unterstützt werden sollen, wurde eine existierende prototypische grafische Benutzungsoberfläche mit zwei unterschiedlichen analytischen Methoden der Softwareevaluation untersucht und Anforderungen für eine alternative Oberfläche abgeleitet. Anschließend wurden beide grafischen Benutzungsoberflächen in einer experimentellen Untersuchung mit zehn berufserfahrenen Personen verglichen.



In this article, the use-centered interface design for an adaptable administration system for chemical process design is presented. On the basis of the main work activities to be supported with the tool, two different analytical evaluation methods were applied to the prototypical original interface. The derived requirements led to the design and implementation of an alternative interface which was compared with the original one in an experimental study with ten users.

Verfahrenstechnische Entwicklungsprozesse lassen sich als komplex, iterativ und kreativ charakterisieren. Im Laufe des Entwicklungsprozesses erstellt, ändert und verwendet das interdisziplinäre Team eine Vielzahl von Dokumenten oder Modellen in unterschiedlichen Softwareprogrammen wie Tabellenkalkulationen, Fließbildwerkzeugen, Simulatoren oder Textverarbeitungssystemen. Gespeichert werden diese Informationen oftmals in Dokumentenmanagement- und Produktmanagement-Systemen. Zusätzlich wird Software zur Koordination der Aufgaben und Entwickler im Sinne des Projektmanagements eingesetzt.

Um eine integrierte Betrachtung der schwach strukturierten Entwicklungsprozesse zu ermöglichen, wurde in einem Teilprojekt des Sonderforschungsbereiches (SFB) 476 „Informatische Unterstützung übergreifender Entwicklungsprozesse in der Verfahrenstechnik“ vom Lehrstuhl für Informatik III (Softwaretechnik) der RWTH Aachen ein neuartiges Softwaresystem entworfen. Das AHEAD (Adaptable and Human-centered Environment for Administration of Development Processes) genannte System besteht aus drei Umgebungen (Westfechtel, 1999). Die Management-Umgebung unterstützt den Projektleiter durch die Zuordnung von Verantwortlichkeiten für Aufgaben und die Über-

wachung von Meilensteinen und Terminen. Die Entwickler-Umgebung informiert darüber, welche Aufgaben zu bearbeiten sind und welche Dokumente dazu benötigt werden. Die Modellierungsumgebung dient schließlich dazu, vordefinierte Aufgabensequenzen und Aufgabentypen zu erstellen.

Im Rahmen der vom Lehrstuhl und Institut für Arbeitswissenschaft durchgeführten Untersuchung wurde nur die Entwickler-Umgebung näher betrachtet. Diese besteht im Prinzip aus zwei Benutzungsoberflächen, und zwar aus der Agenda und dem Arbeitskontext. Ersterer zeigt eine Aufgabenliste mit Informationen zum jeweiligen Projekt, dem Status (nicht bearbeitet, in Bearbeitung u. ä.) und ihrem geplanten Fertigstellungstermin. Bei der Auswahl einer Aufgabe aus der Liste öffnet sich der zugehörige Arbeitskontext (Bild 1 links), der die ein-

zelnen Arbeitsschritte als Aufgabennetz sowie eine Dokumentenliste darstellt. Der dritte Teil der Oberfläche zeigt die verschiedenen Versionen eines Dokumentes als Versionsgraph.

Die Entwickler-Umgebung integriert folglich Konzepte des Projektmanagements mit denen des Dokumenten- und Produktmanagements und stellt daher aus Sicht der Softwaretechnik eine große Herausforderung dar. Um die grundsätzlichen Funktionen der Software überhaupt testen zu können, wurde eine prototypische Benutzungsoberfläche erstellt. Diese stellt den Ausgangspunkt für die Analyse unter software-ergonomischen Gesichtspunkten dar.

Die Entwicklung einer alternativen grafischen Oberfläche folgte einem benutzungszentrierten Ansatz (Flach et al. 1998). Zunächst wurde ermittelt, welche Tätigkeiten eines Entwicklers mit der Software unterstützt werden sollen. Anschließend wurde analytisch untersucht, welche Handlungen und Operationen die ursprüngliche Benutzungsoberfläche zur Durchführung dieser Tätigkeiten erfordert. Die grafische Darstellung erfolgte mit den Diagrammen der Hierarchischen Aufgabenanalyse. Dabei konnte festgestellt werden, dass selbst für einfache Aktionen wie „Erstelle ein neues Dokument“ eine Vielzahl von Mausbewegungen und Klicks notwendig war. Zur Entlastung des Benutzers wurden daher für die alternative grafische Oberfläche unterschiedliche einzelne Funktionen zu einer mit einem Mausklick erreichbaren Funktion zusammengefasst.

Dieser Artikel ist eine gekürzte Version eines Beitrages für die Konferenz HCI International 2003 (Foltz et al., 2003), die im Juni 2003 in Griechenland stattfand.

Weitere Informationen sind von folgenden Projektservern abrufbar: www.iaw.rwth-aachen.de/projekte/sfb476, www-i3.informatik.rwth-aachen.de/research/sfb476/index.html.

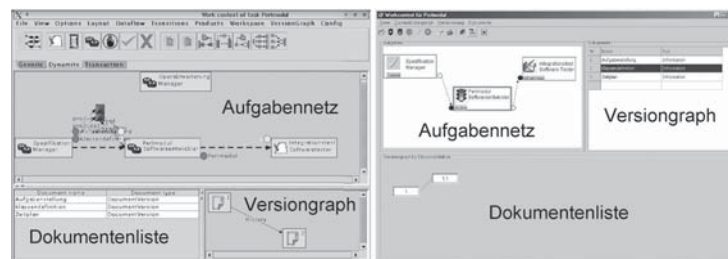


Bild 1: Arbeitskontext des Administrationssystems (links: Ursprüngliche Benutzungsschnittstelle, rechts: Alternative B.)

Parallel dazu konnten im Rahmen einer heuristischen Evaluation weitere Schwachstellen aufgedeckt werden. Dazu zählen beispielsweise die unterschiedliche Bezeichnung der gleichen Funktion in Menüs und auf Schaltflächen oder die wahllose Anordnung von Funktionen, die weder die Reihenfolge noch die Häufigkeit der Benutzung beachtete, d. h. allgemeine ergonomische Kompatibilitätskriterien verletzte.

Auf Basis dieser Erkenntnisse wurde eine alternative Benutzungsoberfläche für die Entwickler-Umgebung entworfen und unter Verwendung von Borlands Delphi als horizontaler Prototyp (Nielsen, 1993) implementiert.

In einer Studie wurden beide Benutzungsoberflächen empirisch untersucht, und zwar aus zwei Gründen: zum einen sollte verglichen werden, wie effizient und effektiv die Anwender mit den beiden grafischen Oberflächen arbeiten, zum anderen wurde erwartet, weitere, in den analytischen Untersuchungen nicht festgestellte, software-ergonomische Mängel aufzudecken.

Der empirische Vergleich wurde mit zehn berufserfahrenen männlichen Versuchspersonen (Alter: 27-34 Jahre, durchschnittliche Berufserfahrung: 2,5 Jahre), von denen acht in der Softwareentwicklung tätig sind, durchgeführt und auf Video aufgezeichnet. Zum Einsatz kam ein Notebook, auf dem beide grafischen Prototypen installiert waren. Nach einer Vorbefragung und Einführung wurden die Versuchsteilnehmer zunächst gebeten, verschiedene Objekte der Benutzungsoberfläche zu benennen.

An Hand von drei gezeigten Bildschirmabzügen sollten die Probanden anschließend die entsprechenden Zustände herstellen und drei kleinere, für den Umgang mit AHEAD typische Aufgaben bearbeiten. Hierbei wurden die Bearbeitungszeiten per Stoppuhr erfasst. Alle Versuchspersonen verwendeten beide Benutzungsoberflächen, wobei die eine Hälfte erst die ursprüngliche und dann die alternative grafische Oberfläche benutzte, während die andere Hälfte in umgekehrter Reihenfolge vorging. Damit dienten die Benutzungsoberflächen

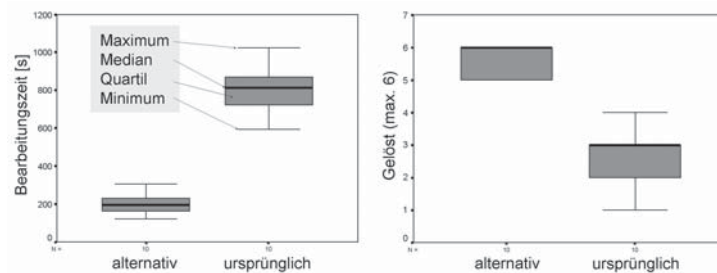


Bild 2: Boxplots zur Bearbeitungszeit und der Anzahl gelöster Zustände und Aufgaben pro Proband bei alternativer und ursprünglicher Entwickler-Benutzungsoberfläche

(ursprünglich vs. alternativ) und die Reihenfolge (ursprünglich-alternativ vs. alternativ-ursprünglich) als unabhängige Variablen. Zur statistischen Auswertung der Untersuchungsergebnisse wurden t-Tests für abhängige Stichproben zum Vergleich zweier Stichprobenmittel verwendet. Die Auswertung ergab, dass die Leistungsmaße der Probanden bei Verwendung der alternativen Benutzungsoberfläche sich signifikant ($p < 0,01$) von den Leistungsmaßen der ursprünglichen Benutzungsoberfläche unterscheiden. Darüber hinaus wurde empirisch belegt, dass dies unabhängig von der Reihenfolge der Darbietung der Fall ist. Bild 2 zeigt exemplarisch einen Vergleich der Bearbeitungszeiten (Effizienzmaß, linke Seite) und der Anzahl der gelösten Zustände und Aufgaben (Effektivitätsmaß, rechte Seite), wobei nicht zwischen den beiden Untersuchungsgruppen unterschieden wird.

Zusätzlich zu den objektiven Messgrößen wie der Bearbeitungsdauer oder dem Lösungsgrad wurde mit Hilfe des IsoMetrics-Fragebogens (Willumeit et al., 1996) eine subjektive Einschätzung der software-ergonomischen Eignung erhoben. Die freien Kommentare der Probanden sowie die Beobachtungen der Versuchsleiter konnten zur Beantwortung konzeptioneller Fragen, die über die reine Oberflächengestaltung hinausgehen, herangezogen werden:

- Das im Arbeitskontext angezeigte Aufgabennetz mit der aktuell bearbeiteten sowie der bzw. den vor- und nachgelagerten Aufgaben wurde in beiden Benutzungsoberflächen häufig als Phasenmodell der zu bearbeitenden Aufgabe und nicht als Ablaufmodell des gesamten Arbeitsprozesses interpretiert.

- In beiden Benutzungsoberflächen wurde nur selten erkannt, dass Dokumente von vorgelagerten Aufgaben explizit vom Benutzer „herunterzuladen“ waren.
- Der Unterschied zwischen den Zuständen „Waiting“ (noch nicht bearbeitet) und „Suspended“ (Bearbeitung unterbrochen) wurde nur selten richtig erkannt.
- Die Dokumentenliste im Arbeitskontext unterstützte in beiden Oberflächen keine Unterscheidung in „selbst erzeugte“ und „als Arbeitsgrundlage zur Verfügung gestellte“ Dokumente.

Zusammenfassend kann festgehalten werden, dass mit Hilfe des benutzungsorientierten Ansatzes effiziente und effektive Benutzungsoberflächen entwickelt werden können. Voraussetzung ist jedoch, dass unterschiedliche analytische und empirische Methoden der Software-Ergonomie eingesetzt werden.

Darüber hinaus muss betont werden, dass das frühzeitige Einbeziehen software-ergonomischer Erkenntnisse schneller zu benutzbaren grafischen Oberflächen führt, als die sequenzielle Bearbeitung, bei der die Softwaretechnik zunächst Funktionen programmiert und erst dann eine Untersuchung der Benutzbarkeit, neudeutsch Usability, durchgeführt wird.

Literatur:

- Flach, J.M.; Tanabe, F.; Monta, K.; Vicente, K.J., Rasmussen, J. (1998) An ecological approach to interface design. In: HFES (Ed.) Proceedings of the Human Factors and Ergonomics Society 42nd Annual Meeting. Santa Monica: HFES, S. 295-299.
- Foltz, C.; Westfechtel, B.; Schmidt, L.; Luczak, H. (2003) Use-Centered Interface Design for an Adaptable Administration System for Chemical Process Design. In: Stephanidis, C.; Jacko, J. (Eds.) Human-Computer Interaction. Theory and Practice (Part II). Volume 2 of the Proceedings of HCI International 2003, June 22-27, 2003, Crete, Greece, Mahwah: Lawrence Erlbaum Associates, S. 365-369
- Nielsen, J. (2003) Usability Engineering. San Diego: Morgan Kaufman.
- Westfechtel, B. (1999) Models and Tools for Managing Development Processes. Berlin: Springer.
- Willumeit, H.; Gediga, G.; Hamborg, K.-C. (1996) IsoMetrics: Ein Verfahren zur formativen Evaluation von Software nach ISO 9241/10. Ergonomie & Informatik, Nr. 27, S. 5-12.