

## STEPS - EINE ORIENTIERUNG DER SOFTWARETECHNIK AUF SOZIALVERTRÄGLICHE TECHNIKGESTALTUNG

Christiane Floyd, Berlin

STEPS steht für "Softwaretechnik für evolutionäre, partizipative Systementwicklung". STEPS ist ein methodischer Ansatz in der Softwaretechnik, der seit 1978 an der Technischen Universität Berlin entwickelt wurde.

Die wichtigsten Bestandteile von STEPS sind:

- eine prozeßorientierte Sichtweise der Softwareentwicklung, die von uns anhand von praktischen Erfahrungen und im Einklang mit unseren Wertvorstellungen als grundlegendes Paradigma für die Softwaretechnik anstelle der herkömmlichen, produktorientierten Sichtweise gesetzt wurde,
- ein menschenzentrierter Qualitätsbegriff, der sich vorrangig am angestrebten Einsatz von Software, und erst darüber vermittelt an Eigenschaften der Software selbst orientiert,
- ein zyklisches Projektmodell, das auf dieser Sichtweise beruht und das die Softwareentwicklung in Zyklen anordnet, die jeweils die Herstellung und Erprobung einer Version zum Gegenstand haben und anschließend zu einer geplanten Revision führen,
- aufeinander abgestimmte Methodenkomponenten für die Schwerpunkte Anforderungsermittlung, Dialogschnittstellenentwicklung und Softwareentwurf sowie zur systematischen Programmierung,
- gestaltbildende Projekttechniken, die die kreative Zusammenarbeit in Teams und die Kommunikation zwischen Entwicklern und Benutzern fördern.

Zur Entstehung von STEPS haben zwei über Jahre hinweg parallel verfolgte Anliegen geführt. Zum einen befaßte ich mich bereits in meiner industriellen Tätigkeit mit der Weiterentwicklung und Erprobung von Methoden im praktischen Einsatz, was mich frühzeitig für den Problemkreis Tauglichkeit von Methoden und Einlösbarkeit von Ansprüchen, die mit Methoden verbunden werden, sensi-

bilisiert hat. Das andere Anliegen war und ist, daß wir Software so entwickeln können wollten, daß sie sich in ihrer Einsatzumgebung als Arbeitsmittel für ihre Benutzer bewähren würde.

Bei STEPS schwebt uns als Anwendungsbereich die Entwicklung von Softwaresystemen vor, die in menschliches Arbeitshandeln eingebettet werden. Man beachte, daß wir damit eine Klassifikation nach Aspekten des Softwareeinsatzes machen. Zu diesem System gehören u. a. Informationssysteme, Textverarbeitungssysteme, Programmierungsumgebungen oder CAD-Systeme.

Zur Perspektive von STEPS gehört zum einen der menschenzentrierte Qualitätsbegriff, festgemacht an dem Anliegen, angemessene Arbeitsmittel für die unterstützten Arbeitsprozesse zu entwickeln. Ich möchte das gegenüber dem herkömmlichen Qualitätsbegriff abgrenzen. In der Regel hat man dabei die Korrektheit bzw. die Effizienz des Produktes im Auge, quasi als Lippenbekenntnis kommt dann noch die sogenannte Benutzerfreundlichkeit dazu. Das bloße Wort "Benutzerfreundlichkeit" macht deutlich, daß man dabei von fertig konzipierten Systemen ausgeht, die man an der Oberfläche, wie man sagt, verzuckert. Wir halten das nicht für ausreichend. Will man DV-Systeme als angemessene Arbeitsmittel für ihre Benutzer konzipieren, so müssen sich die Informatiker im Rahmen der Anforderungsanalyse mit den Arbeitsaufgaben genau auseinandersetzen, und zwar in Abweichung von den klassischen Herangehensweisen nicht nur mit deren formalisierbarem Informationsverarbeitungsanteil, sondern auch mit informellen Aspekten. Wenn man diese Fragen ernst nimmt, so führt das zu den Schwerpunkten in der Softwaretechnik, mit denen wir uns beschäftigen. Ein weiterer wesentlicher Aspekt der Perspektive von STEPS ist die darin verankerte prozeßorientierte Sichtweise.

Die prozeßorientierte Sichtweise betrachtet Software im Zusammenhang mit Lern- und Kommunikationsprozessen von Menschen, und zwar gibt es da unterschiedliche Arten. Software ist Gegenstand von Lern- und Kommunikationsprozessen dann, wenn es darum geht herauszufinden, welche Software man überhaupt braucht. Das tritt beileibe nicht nur in den Anfangsphasen der Softwareentwicklung auf, sondern häufig auch erst nach der Installation, wenn man die realen Anforderungen plötzlich besser weiß. Ferner finden wir Software bei der Benutzung als Unterstützung von Arbeitsprozessen und unter Umständen auch als Rahmenbedingung oder Einschränkung

von Kommunikationsprozessen. Das sind die Rollen, die Software einnehmen kann und sie betreffen Arbeits-, Lern- und Kommunikationsprozesse bei Entwicklern und Benutzern. Dabei sind Softwareentwicklung und -einsatz gemeinsam zu betrachten. Die prozeßorientierte Perspektive führt zu einer Schwerpunktverlagerung, die man auf verschiedene Aspekte beziehen kann.

Bei dieser veränderten Vorgehensweise wird ein tiefgreifendes Umdenken bei der Projektgestaltung in mehrerer Hinsicht impliziert. Zum einen müssen die Bereitwilligkeit und auch die Voraussetzungen für eine partnerschaftliche Zusammenarbeit zwischen Entwicklern und Benutzern vorhanden sein. Das erfordert sowohl entsprechende betriebliche Bedingungen und die Unterstützung des zuständigen Managements, als auch wesentliche menschliche Grundhaltungen und Qualifikationen bei den Entwicklern und auch bei den Benutzern, insbesondere die Fähigkeit und Willigkeit, Kritik in einer konstruktiven Form zu äußern und entgegenzunehmen. In der Praxis muß man darauf gefaßt sein, daß man ein Programm erstellt, dessen Qualität sich erst in der Benutzung erweisen kann. Der Benutzer wird dieses Programm in einer für den Entwickler ganz unerwarteten Weise erleben und unter Umständen mit einer Kritik kommen, auf die man als Entwickler nicht vorbereitet ist. Das ist ein Lernprozeß für beide Seiten.

#### Literaturverzeichnis

- Castner, M. & Pasch, J.(1986): Graphitti - ein graphisch orientierter Schnittstelleneditor für Spezifikation und Entwurf. Proc. der 16. Jahrestagung der GI, Berlin, 6.-10. Oktober 1986. Berlin; Heidelberg; New York; Tokyo: Springer.
- Floyd, C.(1981): A Process-Oriented Approach to Software Development. In: Systems Architecture. Proc. of the 6th European ACM Regional Conf.; Westbury House, S. 285-294.
- Floyd, C.(1983): Grundzüge eines Paradigmenwechsels in der Softwaretechnik. Manuskript eines Vortrags, gehalten an der Humboldt Universität, Berlin im Dezember 1983.
- Floyd, C.(1984): A Systematic Look at Prototyping. In: Budde, R., Kuhlenkamp, K., Mathiassen, L., Züllighoven, H.(Hrsg.): Approaches to Prototyping. Berlin; Heidelberg; New York; Tokyo: Springer, S. 1-18.
- Floyd, C.(1985): On the Relevance of Formal Methods to Software

- Development. In: Ehrig, H., Floyd, C., Nivat, M., Thatcher, J. (Hrsg.): Proc. of the International Joint Conference on Theory and Practice of Software Development (TAPSOFT), Vol. 2: Formal Methods and Software Development. Berlin; Heidelberg; New York; Tokyo: Springer, S. 1-11.
- Floyd, C.(1986): Outline of a Paradigm Change in Software Engineering. Proc. of the International Conference on the Development and Use of Computer-based Systems, Aarhus.
- Floyd, C. & Keil, R.(1983): Softwaretechnik und Betroffenenbeteiligung. In: Mambrey, P., Oppermann, R.(Hrsg.): Beteiligung von Betroffenen bei der Entwicklung von Informationssystemen, Campus, S. 137-164.
- Floyd, C. & Keil, R.(1983): Adapting Software Development for Systems Design with the User. In: Briefs, U., Ciborra, C., Schneider, L.(Hrsg.): Systems Design For, With and By the Users. North-Holland, S. 163-172.
- Floyd, C. & Keil, R.(1984): Integrative Systementwicklung - Ein Ansatz zur Orientierung der Softwaretechnik auf die benutzergerechte Entwicklung rechnergestützter Systeme. BMFT Forschungsbericht DV 84-003.
- Floyd, C. & Pasch, J.(1985): Methoden für den Entwurf großer Softwaresysteme. In: Morgenbrod, H., Remmele, W.(Hrsg.): Entwurf großer Software-Systeme. Berichte des German Chapter of the ACM 19, Stuttgart: Teubner, S. 12-37.
- Floyd, C. u. a. (1986): Arbeitsunterlagen zur Lehrveranstaltung "Einführung in Software Engineering"; gehalten an der TU Berlin im Sommersemester 1986.
- Keil-Slawik, R.(1985): Aufgabenbezogene, dialogorientierte Softwareentwicklung. In: GI Softwaretechnik-Trends, Mitteilungen der Fachgruppe "Software Engineering", Heft 5-2, Juni 1985, S. 105-135.
- Keil-Slawik, R. & Pasch, J.(1987): Methodisches Vorgehen bei der Entwicklung von Dialogschnittstellen. Erscheint in: Proc. des 8. Internationalen Kongresses Datenverarbeitung im Europäischen Raum "Quo vadis EDV? - Realität und Vision 1987", Wien, 30.3.-3.4.1987.
- Prof. Dr. Christiane Floyd, Technische Universität Berlin, Institut für Angewandte Informatik, Franklinstraße 28/29, FR 5-6, 1000 Berlin 10