

Benutzerpartizipation aus Sicht von SW-Entwicklern und Benutzern

Eine Untersuchung von beteiligungsorientierten SW-Entwicklungsprojekten

Astrid Beck

Fraunhofer-Institut für Arbeitswirtschaft und Organisation, Stuttgart

Zusammenfassung

SW-Entwickler müssen sich mit den Arbeitsaufgaben der späteren Benutzer auseinandersetzen, um zu einer benutzungsgerechten Gestaltung zu kommen. Zwischen Systemgestalten und Systemanwendern entstehen jedoch aufgrund verschiedener Zielvorstellungen und unterschiedlicher Voraussetzungen eine Reihe von Konfliktsituationen. Dazu ist im Rahmen des vom BMFT geförderten Arbeit-und-Technik-Projekts TASK eine Untersuchung durchgeführt worden, mit dem Ziel, einerseits SW-Entwicklern und Benutzern mehr Einblick in partizipative Prozesse zu bieten und andererseits mögliche Handlungsalternativen aufzuzeigen, um damit zu besseren Voraussetzungen für den Gestaltungsprozeß computerunterstützter Arbeitsaufgaben zu kommen. Als zentraler Konfliktpunkt ist die mangelnde organisatorische Unterstützung durch das Management vor allem auf Seiten der Benutzer bzw. Fachabteilungen anzusehen. Erschwerend wirkt sich zudem die Unsicherheit der Software-Entwickler aus, die Benutzer geeignet in die Projektarbeit einzubinden.

1 SW-Entwickler im Spannungsfeld von Arbeitsgestaltung und Softwareentwicklung

SW-Entwickler müssen sich mit den Arbeitsaufgaben der späteren Benutzer auseinandersetzen, um zu einer benutzungsgerechten Gestaltung zu kommen. Einerseits sollen die Arbeitsaufgaben angemessen unterstützt werden, andererseits werden die Arbeitsabläufe durch den Einsatz von Software unmittelbar beeinflusst. Dies ist als permanenter Anpassungsprozeß zu verstehen.

Es liegt daher nahe, die Benutzer selbst – als Experten ihrer Arbeit – in den Entwicklungsprozeß miteinzubeziehen: eine Erkenntnis, die sich langsam unter den SW-Entwicklern in der Praxis durchzusetzen beginnt. Das Software-Entwickler Benutzerbeteiligung nicht (mehr) ablehnend gegenüberstehen, konnten bereits Peschke [5] und Spinaz und Waeber [7] in ihren Untersuchungen bestätigen, die beide von 70% Zustimmung ausgehen. Zum organisatorischen Umfeld von Beteiligung oder zu Prototyping gibt es in diesen Untersuchungen leider wenig Aussagen.

Es geht also weniger darum, Überzeugungsarbeit zu leisten, daß Partizipation der Benutzer sinnvoll und notwendig ist, als vielmehr um praktikable Vorgehensweisen, die Partizipation erfolgreich realisieren helfen. Der Beteiligungsprozeß als solcher läuft jedoch bisher in den wenigsten Fällen für die Beteiligten zufriedenstellend ab. Zwischen Systemgestaltern und Systemanwendern entstehen aufgrund verschiedener Zielvorstellungen und unterschiedlicher Voraussetzungen eine Reihe von Konfliktsituationen. Welcher Art sind diese Konflikte konkret und wie kann man sie aus Sicht der Betroffenen in den Griff bekommen? Wie denken SW-Entwickler und Benutzer übereinander? Was sind die gegenseitigen Anforderungen und Ansprüche? Welche Erfahrungen aus beteiligungsorientierten Projekten und zu Prototyping gibt es bereits? Kann man daraus ein Vorgehen für ein erfolgversprechendes, praktikables Partizipationsmodell ableiten?

Zu den genannten Fragestellungen ist eine Untersuchung durchgeführt worden, mit dem Ziel, einerseits SW-Entwicklern und Benutzern mehr Einblick in partizipative Prozesse zu bieten und andererseits mögliche Handlungsalternativen aufzuzeigen, um damit zu besseren Voraussetzungen für den Gestaltungsprozeß computerunterstützter Arbeitsaufgaben zu kommen. Für die Beteiligung der Benutzer bei Systementwurf und -gestaltung und bei der Um- oder Neugestaltung ihrer Arbeitstätigkeit gibt es nach wie vor kein Patentrezept (vgl. z.B. Floyd, Keil [1] und Wicke [9]). Ziel der Untersuchung war es daher, eine Reihe von Anforderungen an den Beteiligungsprozeß und ein geeignetes Vorgehen zu stellen. Hier sollte verstärkt auch auf die Sichtweisen der beteiligten Benutzer eingegangen werden. Es sollten vor allem Aussagen zur organisatorischen Einbettung von Beteiligung (s.a. Jansen, Schwitalla, Wicke [3] und Schubert, Zink [6]), zu Methoden- und Werkzeugeinsatz, zu Prototyping und zu Qualifizierung gewonnen werden, was in früheren Untersuchungen nicht thematisiert wurde (z.B. Klutmann [4]).

Im folgenden wird zunächst die im Rahmen des TASK-Projekts durchgeführte Untersuchungsmethode vorgestellt, daran anschließend werden die Ergebnisse vorgestellt. Abschließend werden Anforderungen an eine kooperative Systemgestaltung formuliert.

2 Die Untersuchung: Vorgehen und Methode

Für die Gespräche mit den SW-Entwicklern und Benutzern wurde ein Gesprächsleitfaden entwickelt. Der Leitfaden enthält einige Fragen mit Antwortkategorien sowie weitgehend offen formulierte Fragen zu folgenden Untersuchungsschwerpunkten:

- Fragen zur Person (u.a. Ausbildung, Projekterfahrung)
- Fragen zum Projekt (u.a. Laufzeit, Zahl der Mitarbeiter, Zahl der Benutzer)

- Projektvorphase (u.a. Projektinitialisierung, Informationsflüsse, Einbindung von Betriebsrat, erster Kontakt)
- Organisation der Beteiligung (u.a. Form der Mitwirkung, Rechte, Rollen, Verantwortlichkeiten, Zusammensetzung der Projektgruppe, Projektvertrag)
- Qualifizierung (SW-Entwickler und Benutzer, Art, Inhalt)
- Auswahl der Benutzer (Art, Freistellung, erzwungen/freiwillig, Kontrolle)
 - Zeitaufwand
 - Beteiligung und Phasen
- Methoden- und Werkzeugeinsatz (u.a. SW-ergonomische Prinzipien, Darstellungsmethoden, CASE-Tools)
- Wissen des Benutzers über SW-Entwicklung (Methoden, Vorgehen)
- Wissen des Entwicklers über Fachaufgabe (Methoden, Vorgehen)
- Prototyping (Phasen, Art, Dokumentation, Vorgehen, mögliche Ziele)
- Kommunikation und Kooperation (u.a. Projekthilfsmittel)
- Situation nach dem Projekt (Betreuung, Änderungen)
- Einschätzung von Partizipation und Vorschläge

Einige der Fragen sind als Behauptung formuliert und sollten auf einer dreistufigen Ratingskala als “stimmt/sinnvoll/wird praktiziert”, “unentschieden” oder “stimmt nicht/ nicht sinnvoll/geht nicht” beurteilt werden.

Es wurde sowohl für die Gespräche mit den Entwicklern als auch mit den Benutzern ein identischer Leitfaden eingesetzt. Lediglich die Frage zum Wissen über das Vorgehen der Benutzer/Entwickler und die Behauptungen über die Benutzer/Entwickler wurden spezifisch an die jeweilige Gruppe formuliert. Die Gespräche wurden anhand des Leitfadens handschriftlich protokolliert. Die Gesprächspartner wurden zuvor über Ziel und Zweck der Untersuchung informiert, und es wurde ihnen absolute Anonymität zugesichert; ein entsprechendes Infoblatt wurde ihnen ausgehändigt. Das Ziel war, von Anfang an eine offene, vertrauensvolle Atmosphäre zu schaffen; es sollte ein Gespräch geführt werden — kein Interview im Sinne einseitigen Ausfragens.

Es wurden nur mit direkt beteiligten Mitarbeitern Gespräche geführt und zwar möglichst aus jedem Projekt mit mindestens einem SW-Entwickler und einem Benutzer. Damit war es möglich, von beiden Seiten die spezifischen Sichtweisen zu bestimmten Projektsituationen zu erhalten. Es wurden 10 Gespräche mit SW-Entwicklern (davon 1 Entwicklerin) und 11 Gespräche mit Benutzern (davon 2 Benutzerinnen) geführt. Die Gespräche dauerten 1 bis 1,5 Stunden.

Die SW-Entwickler sind entweder in SW-Häusern oder in Unternehmen mit eigener DV-Abteilung tätig. Bei den Benutzern handelt es sich entsprechend um die

Personen, die beim auftraggebenden Kunden mit der zu entwickelnden Software arbeiten werden oder um die späteren Benutzer in den Fachabteilungen. Untersuchungsfeld waren 9 Projekte aus den Bereichen Büro, Versicherungswesen, Pharmaindustrie, TV-Branche sowie Produktionssteuerung. Ausgewählt wurden die Projekte über persönliche Kontakte, wobei darauf geachtet wurde, daß sowohl Softwarehäuser als auch Unternehmen mit eigener DV-Abteilung mit Projekten von unterschiedlicher Laufzeit und Größe in die Untersuchung aufgenommen wurden. Die Projekte hatten eine durchschnittliche Laufzeit von 3,8 Jahren. Ebenso wurde ein breites Anwendungsfeld mit Benutzern verschiedener Berufsgruppen angestrebt. Es wurden keine Projekte betrachtet, die selbst in irgendeiner Form sich verstärkt (z.B. durch finanzielle Förderung) mit Benutzerpartizipation auseinandersetzen.

Die Gespräche wurden anschließend anhand der Themenstellungen und der aufgedeckten Problemkomplexe ausgewertet.

3 Ergebnisse

Generell wurde die Beteiligung der späteren Benutzer am Software-Entwicklungsprozeß von den SW-Entwicklern als notwendig gesehen. Wenngleich einem Entwickler der Prozeß der Partizipation als insgesamt sehr aufwendig erschien (Zitat: "aber da muß man halt durch...") und einem anderen die Benutzer nicht genügend kompetent erschienen (er sprach oft vom DAU — dem "Dümmbsten Anzunehmenden User"), so wurde doch von allen einhellig die Meinung vertreten, daß nur durch die Einbindung der späteren Benutzer die Software sich angemessen gestalten läßt und dafür das Fachwissen der Anwender benötigt wird.

3.1 Defizite in der Organisation von Beteiligung

Vorgefundene Formen der Beteiligung

Die Projekte wurden durch Kundenauftrag oder durch Auftrag aus der Fachabteilung veranlaßt. Eher überraschend ist die Erkenntnis, daß die beteiligten Benutzer sehr früh informiert wurden und in allen Projekten von Anfang an eingebunden waren. Die Form der Mitwirkung ist als überwiegend passiv einzuordnen. Protokolle und Moderation von Sitzungen liegt immer in der Hand der SW-Entwickler. Es wurde aber mehrheitlich der Vorschlag begrüßt, diese Aufgaben rotieren zu lassen. Die Projektgruppe setzte sich in allen Fällen aus SW-Entwicklern und (zeitweilig) den Benutzern zusammen. Im Projektvorfeld verhandelte immer die Leitung der Fachabteilung mit den SW-Entwicklern, hier waren noch keine Benutzer dabei. In keinem der untersuchten Projekte fand eine Einbindung des Betriebsrats statt.

Fehlender Projektvertrag

Als sehr wichtig wurde der Projektvertrag bezeichnet, der die Kompetenzen aller Beteiligten und ihre zeitliche und organisatorische Einbindung regelt. Alle Gesprächspartner behaupteten, einen entsprechenden Vertrag gäbe es für ihr Projekt. Auf Nachfrage mußte auf den Unterschied von Kundenauftrag oder Pflichtenheft und dem Projektvertrag, der die Verbindlichkeiten der Gruppenmitglieder untereinander regelt, hingewiesen werden. Einen Vertrag entsprechender Art gibt es aber bisher noch bei keinem der untersuchten Projekte.

Fehlende zeitliche Regelungen

Über die zeitliche Einbindung der Benutzer im Projekt gibt es durchweg nur vage Absprachen zu Projektbeginn, welche im laufenden Projekt jedoch nicht eingehalten werden. Das zentrale Problem ist oft, daß erfahrene, im Entwicklungsprojekt engagierte Benutzer auch in ihrer eigentlichen Tätigkeit meist stark eingebunden sind. Dann werden Absprachen über den zeitlichen Rahmen der Projektstätigkeit oft hinfällig, da selten für einen Personalausgleich beim Anwender gesorgt wurde. Drei Benutzer gaben an, die Mehrbelastung nur durch ständige Überstunden bewältigen zu können.

Unklare Kompetenzen und fehlende Abstimmungsprozesse

Acht Benutzer wünschten sich klarere Regelungen zu ihren Rechten und Verantwortlichkeiten. Zwei Benutzer würden gerne bei der Hardwareauswahl beteiligt sein. Bei auftretenden Problemen und neuen Anforderungen fehlte den meisten Benutzern die Möglichkeit zur Mitwirkung, da "solche Sachen immer von oben entschieden" würden. Entscheidungen fallen oft — nach Meinung der Benutzer — falsch aus. Resultierende Konflikte werden als hinderlich auch bei den Entwicklern gesehen, die bemerken, daß die Motivation zur Mitarbeit der Benutzer durch Fehlentscheidungen deutlich sinkt.

Mangelnde Kooperation und Kommunikation der Benutzer untereinander

Eine Kooperation der Benutzer untereinander findet eher nicht statt. Beteiligte Benutzer informieren ihre Fachkollegen selten oder überhaupt nicht über die Projektarbeit und den -fortschritt. Die Fachkollegen reagieren nicht selten mit Desinteresse oder Neid. Des öfteren äußern Kollegen Unmut, wenn die beteiligten Benutzer aufgrund von Projektsitzungen zeitweilig in der Abteilung nicht verfügbar sind.

Unsicherheit beim Umgang mit Konflikten

Von allen Teilnehmern wird bestätigt, daß die Zusammenarbeit nicht nur durch den Systementwicklungsprozeß an sich, sondern vor allem stark durch soziale Prozesse geprägt ist. Diese sind gekennzeichnet durch Konflikte, Machtansprüche, wechseln-

de Motivation und unterschiedliche Interessenlagen. Soziale Prozesse werden in der Projektarbeit jedoch selten aus- oder angesprochen, dadurch werden Konflikte ignoriert oder sie verhärten sich. Regelabsprachen, wie beispielsweise bei Unstimmigkeiten zu entscheiden ist, gibt es in keinem der Projekte.

Übereinstimmend wird von den Benutzern, aber auch von einigen SW-Entwicklern beklagt, daß das Management der Fachabteilung für die hier dargestellten Probleme hauptsächlich die Verantwortung trage. Bis auf lose Absprachen und Versprechungen gibt es keine konkreten (schriftlichen) Regelungen. Das Management der Fachabteilung schätze zudem den zeitlichen und fachlichen Aufwand für die Beteiligung viel zu gering ein, schafften keinen Ausgleich und seien im wesentlichen der Meinung, daß "doch eigentlich die SW-Entwickler die Arbeit machen sollen".

3.2 Mangelnde Motivation der Benutzer

Als zentrales Problem wurde vor allem die Motivation der beteiligten Benutzer von den SW-Entwicklern genannt. Entweder seien sie mehr oder weniger gegen eine Einführung neuer Hard- und Software oder sie arbeiteten nicht so mit, wie sich das die EDV-Entwickler vorstellen. Hier ist stark die Motivation und Kompetenz der Beteiligten angesprochen.

Über mögliche Veränderungen der Tätigkeit erhalten die Benutzer keine oder nur unzureichende Informationen. SW-Entwickler klagen darüber, daß Benutzer meist gar keine Veränderungen wünschen bzw. zu stark an den alten Strukturen und Vorgehensweisen festhalten. Ein Benutzer urteilte über seine nicht beteiligten Kollegen: "Ideen haben sie alle, vor allem hinterher, nur mitarbeiten — das will keiner".

Die Auswahl der Benutzer funktioniert in fast allen Projekten "von oben herab". So geben über die Hälfte der Benutzer an, daß ihre Mitarbeit im Projekt eher erzwungen ist. Das die Benutzer oft nur widerwillig im Projekt mitarbeiten wurde auch so von den SW-Entwicklern erkannt.

Es läßt sich beobachten, daß bei kleineren Projekten der Kontakt und die Zusammenarbeit zwischen den Mitarbeitern deutlich besser ist. Je größer die Projekte, desto größer wird die Kluft zu anderen Projektmitarbeitern. Was andere machen, ist nicht mehr transparent, insbesondere, wenn Mitarbeiter noch in anderen Projekten mitarbeiten, was bei über der Hälfte der Benutzer und der SW-Entwickler der Fall ist. Entsprechend fehlt ein gemeinsames Gruppenbewußtsein, ein "Wir"-Gefühl und damit ein entscheidender Motivationsfaktor.

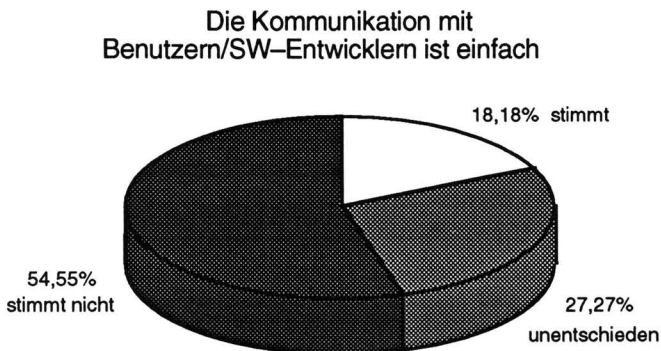
In zwei Projekten wurden gemeinsame Feiern bei Phasenabschlüssen und gemeinsame Ausflüge als besonders fördernd für die Motivation angegeben. In einem Projekt gingen entsprechende Aktivitäten von den Entwicklern, im anderen von den Benutzern aus. Auffallend ist bei beiden Projekten, daß die Benutzer auch im Arbeitsprozeß stärker integriert sind, d.h. daß sie mehr Aufgaben im Beteiligungsprozeß haben (Mitarbeit bei Anforderungsermittlung, Handbucherstellung und Testkonzept) und sich zufriedener über die Zusammenarbeit äußern. Sowohl die Entwickler, als auch die Benutzer dieser Projekte stimmen der Behauptung "Die Benutzer sind zur Mitarbeit motiviert" zu.

3.3 Widersprüchliche Erwartungen der Beteiligten

Die gegenseitigen Erwartungen der SW-Entwickler und Benutzer sind als stark widersprüchlich einzuschätzen. So stimmen fast alle SW-Entwickler den Aussagen "Die Benutzer erwarten sich zuviel von der SW-Lösung" und "Die Benutzer erwarten das nächste Ergebnis zu schnell" zu, während die Benutzer diese bestreiten. Die SW-Entwickler entdecken bei den Benutzern "1000 Wünsche und überzogene Ideen", denen sie aber nicht nachkommen könnten, da sie "einfach unsinnig" oder "am alten System" orientiert seien. Die Benutzer wüßten nicht, was sie wollen und würden in Projektsitzungen ihre eigenen Probleme stundenlang diskutieren.

Benutzer sind mit den SW-Entwickler unzufrieden, wenn diese sie mit Aussagen konfrontieren würden wie: "Das ist technisch nicht machbar", "Das kann unser Tool nicht", "das dauert zu lange/ist zu umständlich".

Betrachtet man Benutzer und Entwickler gemeinsam, so können über 80% nicht der Aussage zustimmen, daß die Kommunikation miteinander einfach sei.



3.4 Unzureichende Qualifizierung der Beteiligten

Eines der Hauptprobleme bei der Realisierung von Beteiligung liegt in der oft fehlenden Qualifikation der am Prozeß Beteiligten. Auf Seiten der Software-Entwickler fehlt die Kompetenz, Arbeitsprozesse mit den Benutzern zu gestalten. Sie können dem Auftraggeber auch vorab keinen Überblick vermitteln, in welcher Weise sie die Zusammenarbeit mit den Benutzern gestalten wollen. Auf Seiten der Benutzer fehlt meist das Grundwissen über Möglichkeiten und Probleme der EDV-Gestaltung und des EDV-Einsatzes.

Nur in einem Projekt wurde die später zu verwendende Hardware bereits vorab installiert, damit sich die Benutzer mit ihr vertraut machen konnten.

Auf die an die Benutzer gerichtete Frage, ob sie wüßten, wie SW-Entwickler im SW-Entwicklungsprozeß vorgehen, was deren Aufgaben, Verpflichtungen oder Probleme mit unzureichenden Werkzeugen etc. sind, gibt der Großteil an, keine oder nur geringe Vorstellungen zu haben. Die SW-Entwickler wiederum behaupten, über die Fachaufgaben der Benutzer gut informiert zu sein. Allerdings stimmt nur ein SW-Entwickler der Behauptung zu, "Die Benutzer können ihr eigenes Fachgebiet und ihre Arbeitsaufgaben gut erläutern". Es stellt sich somit die Frage, wie die SW-Entwickler zu dem Wissen über die Arbeitsaufgaben kommen und von welcher Qualität entsprechendes Wissen ist.

3.5 Mangelhafter Methoden- und Werkzeugeinsatz

Für die Form der Kommunikation und Zusammenarbeit mit den Benutzern geben die Entwickler mehrheitlich Interview, Workshops/Sitzungen und Bewertung von Dokumenten an.

Als Darstellungsmittel werden verschiedene Datenflußbeschreibungstechniken, Kärtchentechnik/Metaplan, Hardcopies von Bildschirmmasken verwendet.

Auf SW-ergonomische Prinzipien angesprochen, geben fast alle SW-Entwickler, aber keiner der Benutzer an, sie seien ausreichend informiert. In einem Unternehmen wird ein SW-ergonomischer Style Guide eingesetzt, der zum Teil auch den Benutzern bekannt war. Ein eher unzulänglicher Dialogentwurf, der in einigen Projekten anzutreffen war, wird oft damit begründet, die Fachabteilung hätte genau das aber so gewollt bzw. für gut befunden.

CASE-Tools werden in allen Projekten zur Dokumentation oder zur Projektplanung eingesetzt, aber die erstellten Dokumente werden in der Regel nicht für die Benutzer-Entwickler-Kommunikation verwendet.

Die gegenwärtige Situation ist gekennzeichnet durch einen schlechten Informationsstand über den aktuellen Projektfortgang und durch unzureichende Dokumente (fehlend oder zu umfangreich). In keinem der Projekte gibt es allgemein zugängliche Projektunterlagen, geäußert wurde dazu der Wunsch nach elektronischem Informationsaustausch.

Ein eigenes Projektbüro, das jederzeit für alle Projektmitarbeiter zugänglich war, hat nur ein Projekt. Das Projektteam mußte jedoch die an der Wand angebrachten Ablaufpläne wieder abnehmen, da dies nicht dem Erscheinungsbild des Unternehmens entsprach (Kunstgrafiken an weißen Wänden).

3.6 Falsch verstandenes Prototyping

Überraschend ist, daß der Begriff "Prototyping" allen Gesprächspartnern geläufig ist und mehrheitlich behauptet wird, Prototyping würde in ihrem Projekt durchgeführt. Sowohl SW-Entwickler als auch Benutzer sind sich über Ziele und Vorgehensweisen für Prototyping allerdings im unklaren. Die Vorstellungen, was Prototyping sei, reichen von mit einem Zeichenprogramm erstellten Maskenentwürfen bis hin zu voll lauffähiger Software. Prototypen werden nicht oder unzureichend dokumentiert.

Mit Prototyping wird begonnen, obwohl die benötigten Informationen aus der Anforderungsanalyse nicht oder schlecht aufbereitet zur Verfügung stehen. Es läßt sich feststellen, daß die entstehenden Prototypen vielfach nicht den Anforderungen der Aufgabe entsprechen, was wiederum von den Benutzern selten erkannt wird, da sie sich eher auf die Kritik der reinen Benutzungsoberfläche beschränken.

Die immer wieder zitierten Verständnisschwierigkeiten zwischen Entwicklern und Benutzern (z.B. Strohm [8]) können zwar im Prinzip auch in dieser Untersuchung bestätigt werden, sind aber nicht als zentraler Konfliktpunkt anzusehen. Es hat sich vielmehr gezeigt, daß sich Schwierigkeiten bei der Beteiligung durch die mangelnde organisatorische Unterstützung durch das Management vor allem auf Seiten der Benutzer bzw. Fachabteilungen ergeben. Erschwerend wirkt sich zudem die unklare Einstellung der Software-Entwickler aus, die oft unsicher sind, wie die Benutzer geeignet in die Projektarbeit einzubinden sind.

4 Anforderungen an eine kooperative Systemgestaltung

Zusammenfassend lassen sich basierend auf den Ergebnissen der Untersuchung von beteiligungsorientierten SW-Entwicklungsprojekten eine Reihe von Anforderungen

formulieren, die den Entwicklern und Benutzern Anhaltspunkte bieten können, geeignete Vorgehensschritte für erfolgreiche Beteiligung frühzeitig zu identifizieren und umzusetzen. Im folgenden werden die Anforderungen aufgezeigt, die die Defizite überwinden helfen sollen, die sich in der Untersuchung als zentral herausgestellt haben:

1. Die mangelnde organisatorische Unterstützung durch das Management der Fachabteilung.
2. Die Unsicherheit der Software-Entwickler, Benutzer geeignet in die Projektarbeit einzubinden.

4.1 Anforderungen an das Management der Fachabteilung

Eine passive Einbindung der Benutzer in die Projektarbeit ist nicht ausreichend, vielmehr ist deren aktive, eigenverantwortliche Mitarbeit anzustreben. Generell sollte den Benutzern mehr Verantwortung, Einflußmöglichkeit und Entscheidungskompetenz zukommen. Hierzu sind eine Reihe von Voraussetzungen zu schaffen. Auf Seiten des Managements der Fachabteilung sind zu fordern:

- Anreize, die die Benutzer motivieren, Software mitzugestalten (finanzielle Anreize, zeitlicher Ausgleich, Aufstiegsmöglichkeiten etc.),
- Mitspracherecht bei der Auswahl der Mitarbeiter, die beteiligt werden sollen,
- geregelte Freistellung für die Dauer der Projektmitarbeit,
- Qualifizierung der Benutzer (DV-technische Kenntnisse, software-ergonomisches und arbeitswissenschaftliches Wissen, Kenntnisse von Gruppenprozessen, (s.a. S. Hacker, B. Müller-Holz auf der Heide, G. Aschersleben [2]),
- Definition von Pflichten und Verantwortlichkeiten,
- Stärkung der Position gegenüber Entwicklern und Kollegen der Fachabteilung,
- organisatorische Maßnahmen, die es ermöglichen, daß die Benutzer untereinander in einen verstärkten Kooperationsprozeß treten können (z.B. regelmäßige Gruppenbesprechungen),
- aktive Informationspolitik (Unternehmensziele, Strategien, neue Hardware/Software etc.),
- kooperative und demokratische Entscheidungs politik.

4.2 Anforderungen an die Software-Entwickler

Auf Seiten der Software-Entwickler ist das Verständnis um Beteiligungsprozesse zu erweitern. Software-Entwickler müssen zunächst selbst genau wissen, wie Beteiligung aussehen soll, damit sie ihr Verständnis an die beteiligten Benutzer weitergeben können. Dazu gehört eine umfassende und aktive Informationspolitik über Ziele und

Zeitraumen des Projekts. Das geplante Vorgehen ist frühzeitig transparent zu machen.

Umfang und Inhalt der Entscheidungsbefugnisse aller Beteiligten sollten von Anfang an in einem Projektvertrag festgeschrieben werden. Im Projektvertrag ist festzulegen:

- Zusammensetzung der Projektgruppe (Beteiligte, Einbindung vom Betriebsrat),
- die zeitliche Verfügbarkeit und Freistellung von Mitarbeitern,
- Aufgabenverteilung und Verantwortlichkeiten (Rollen, Rechte und Pflichten),
- einzusetzende Methoden, Vorgehensweisen und Werkzeuge,
- Regeln für die Projektarbeit.

Vorab können sich auch für die Software-Entwickler Qualifizierungsmaßnahmen als notwendig erweisen. Kenntnisse, die zu vermitteln sind, umfassen:

- Prototyping als Möglichkeit der Benutzerbeteiligung,
- Fachwissen (aufgaben- und fachbezogene Qualifizierung),
- software-ergonomisches und arbeitswissenschaftliches Wissen,
- Kenntnisse von Gruppenprozessen.

Verwendete Methoden und Werkzeuge in der SW-Entwicklung sollten für die Benutzer transparent sein (u.a. SW-ergonomische Style Guides, Darstellungsmethoden, CASE-Tools), wenn diese Auswirkungen auf die Projektarbeit und die Dokumentation haben.

Nach Möglichkeit sind Hilfsmittel zur Kommunikation und zum verbesserten Dokumentenaustausch zu schaffen, wie z.B. DV-gestützte und für die Projektmitarbeiter immer zugängliche Projektdokumentation (Aufgaben- und Zeitpläne, Projektnachrichten, Glossar, Projektergebnisse, Pflichtenheft, Prototyp etc.). Ein gemeinsames Projektbüro kann Ort für Besprechungen, Standort von Unterlagen und des installierten Prototypen sein.

Mit Prototyping sollte erst dann begonnen werden, wenn klar ist, was genau Umfang und Inhalt des Prototypen sein soll und welche Aufgaben unterstützt werden sollen. Hier sollte ein aufgabenorientiertes Prototyping ansetzen, daß sich an den Arbeitsabläufen der Benutzer orientiert und bei dem die Ziele und Vorgehensschritte von Benutzern und Entwicklern gemeinsam festgelegt werden.

5 Literatur

- [1] Floyd, C. ; Keil, R. (1983): Softwaretechnik und Betroffenenbeteiligung. In: Mambrey, P. ; Oppermann, R. (Hrsg.): Beteiligung von Betroffenen bei der Entwicklung von Informationssystemen, Frankfurt: Campus, 137-164

- [2] Hacker, S. ; Müller-Holz auf der Heide, B. ; Aschersleben, G. (1991): Prototyping in einem Designteam: Vorgehen und Erfahrungen bei einer Software-Entwicklung unter Benutzerbeteiligung. In: Frese, M. ; Kasten, C. ; Skarpelis, C. ; Zang-Scheucher, B., Hrsg.: Software für die Arbeit von morgen: Bilanz und Perspektiven anwendungsorientierter Forschung, Tagungsband, Berlin: Springer-Verlag
- [3] Jansen, K. D. ; Schwitalla, U. ; Wicke, W. (1988): Organisatorische und methodische Aspekte einer beteiligungsorientierten Entwicklung computergestützter Arbeitssysteme, Zwischenbericht zum Forschungsprojekt "Methodisches Instrumentarium zur Partizipation von Arbeitnehmern bei der Entwicklung und Einführung computergestützter Arbeitssysteme"
- [4] Klutmann, B. (1987): Benutzer-Entwickler-Kommunikation im Softwareentwicklungs-Prozeß. In: Schönplüg, W. ; Wittstock, M., Hrsg. (1987): Software-Ergonomie '87, Bericht des German Chapter of the ACM, Band 29, Stuttgart: Teubner-Verlag
- [5] Peschke, H. (1986): Betroffenorientierte Systementwicklung. Prozess und Methoden der Entwicklung menschengerechter Informationssysteme, Frankfurt: Verlag Peter Lang
- [6] Schubert, H.-J. ; Zink, K.J. (1990): Partizipation - Psychologische Grundlagen eines Leitprinzips von Arbeits- und Organisationsgestaltungsmaßnahmen. In: Zeitschrift für Arbeitswissenschaft, 44, 2
- [7] Spinass, P. ; Waerber, D. (1991): Benutzerbeteiligung aus der Sicht von Endbenutzern, Softwareentwicklern und Führungskräften mit Beteiligungserfahrung. In: Ackermann, D. ; Ulich, E. (Hrsg.): Software-Ergonomie '91, Bericht des German Chapter of the ACM, Band 33, Stuttgart: Teubner-Verlag
- [8] Strohm, O. (1991): Projektmanagement bei der Software-Entwicklung, Eine arbeitspsychologische Analyse und Bestandsaufnahme. In: Ackermann, D. ; Ulich, E., Hrsg. (1991): Software-Ergonomie '91, Bericht des German Chapter of the ACM, Band 33, Stuttgart: Teubner-Verlag
- [9] Wicke, W. (1988): Methoden der Partizipation bei der Entwicklung computergestützter Arbeitssysteme. In: Kießler, L., Hrsg.: Computer und Beteiligung, Opladen: Westdeutscher Verlag

Astrid Beck

Institut für Arbeitswissenschaft und Technologiemanagement (IAT)

Fraunhofer-Institut für Arbeitswirtschaft und Organisation (FhG-IAO)

Universität Stuttgart

Senefelderstr. 26

7000 Stuttgart 1

E-Mail: A_BECK@iao.fhg.de