

Systemgestaltung mit Aufgabennetzen

Reinhard Keil-Slawik, Berlin

Kurzfassung

Die Angemessenheit von Software bezüglich der Arbeitsaufgaben der Benutzer ist ein entscheidendes Qualitätsmerkmal. Zur Gestaltung eines angemessenen DV-Systems ist eine aufgabenbezogene Anforderungsermittlung erforderlich, die durch geeignete Modellierungs- und Darstellungstechniken unterstützt wird. Bei der Entwicklung industriell eingesetzter interaktiver Systeme zeigt sich jedoch, daß die meist in den Vordergrund gestellten Anforderungen an die formale Fundierung solcher Techniken in der Praxis nur eine untergeordnete Rolle spielen. Dagegen stehen diejenigen Aspekte im Vordergrund, die eine angemessene Unterstützung der Entwicklungsprozesse bieten.

1 Einleitung

Softwareentwicklung ist Teil eines umfassenderen Systementwicklungsprozesses. Die Systementwicklung beinhaltet tiefgreifende Veränderungen in der Benutzerorganisation und berührt alle Aspekte des menschlichen Arbeitshandelns. Sie betreffen u.a. die Produktions- und Entscheidungsstrukturen, die Art und Anzahl der vorhandenen Arbeitsplätze, die Arbeits- und Kommunikationssituation der Beschäftigten sowie die Organisationsstruktur insgesamt. Techniken und Methoden zur Entwicklung interaktiver Systeme müssen dieser Situation Rechnung tragen.

Um den damit verbundenen Anforderungen nachzukommen, wurde an der TU Berlin der Methodenrahmen STEPS¹ entwickelt, der in kooperativer Arbeit von den Mitgliedern der Forschungsgruppe Softwaretechnik jeweils aufgrund ihrer Erfahrungen in Projekten und Lehrveranstaltungen modifiziert und erweitert wird. STEPS steht für Softwaretechnik für evolutionäre partizipative Systementwicklung und umfaßt u.a. Komponentenmethoden für die aufgabenbezogene Anforderungsermittlung, die Gestaltung der Benutzungsschnittstelle und den Entwurf. Diese werden jeweils durch geeignete Modellierungs- und Darstellungstechniken unterstützt. STEPS verkörpert eine grundlegende Systementwicklungsphilosophie und ist nicht auf diese speziellen Methoden und Techniken festgelegt. Je nach Erfordernis können im Rahmen von STEPS verschiedene Verfahren eingesetzt werden.

Der vorliegende Beitrag beschäftigt sich mit der aufgabenbezogenen Anforderungsermittlung mit Hilfe von Aufgabennetzen. Neben einer kurzen Beschreibung dieser Methodenkomponente werden die Erfahrungen, die mit ihrem Einsatz im Rahmen industrieller Softwareentwicklungsprojekte gemacht wurden, systematisch aufbereitet.

2 Aufgabenbezogene Anforderungsermittlung

Zunächst geht es jedoch darum, die aufgabenbezogene Anforderungsermittlung und ihre Einbettung in die Systementwicklung zu skizzieren, damit die im Anschluß daran beschriebenen Auswertungen und Erfahrungen nachvollziehbar sind. Die Darstellung reflektiert den gegenwärtigen Erkenntnisstand und weicht daher in einigen Punkten von früheren Veröffentlichungen ab.

¹Einen Überblick gibt Floyd (1986)

2.1 Stellenwert der Anforderungsermittlung

Anforderungen stecken den Rahmen für die Systementwicklung ab; ihnen kommt daher eine Schlüsselstellung zu. Wesentlich dabei ist der Zusammenhang zwischen Anforderungen an das DV-System und der Einpassung des Systems in das menschliche Arbeitshandeln. Zur genaueren Bestimmung werden Anforderungen in die nachfolgenden Kategorien unterteilt:

- **Funktionelle Anforderungen** beschreiben das Ein-/Ausgabeverhalten des DV-Systems, d.h. welche Eingaben das System erwartet, welche Listen erzeugt werden sollen usw.
- **Leistungsanforderungen** beziehen sich auf die zur Verfügung stehenden Ressourcen und die optimale Ausnutzung der Betriebsmittel; dazu gehört die Angabe von Mengengerüsten.
- **Handhabungsanforderungen** betreffen das Zusammenwirken von Mensch und Rechner; sie beziehen sich beispielsweise auf die Gestaltung von Bildschirmmasken, die Dialogführung, die Fehler- und Ausnahmebehandlung sowie auf die Gerätebedienung.
- **Einbettungsanforderungen** berücksichtigen spezifische Gegebenheiten einer Organisation, die Randbedingungen für die Software-Entwicklung liefern. Das kann sowohl den Bezug auf bereits vorhandene Hardware und Software beinhalten, als auch die Forderung nach Einhaltung von Verfahrensvorschriften, Standards und Programmierkonventionen sowie Einführungsmodalitäten und Schulungsmaßnahmen.

Im Rahmen der Anforderungsermittlung müssen diese Angaben erarbeitet und für die darauf aufbauende Systemgestaltung bereit gestellt werden.

Bei der aufgabenbezogenen Anforderungsermittlung soll einerseits den Benutzern frühzeitig die Einflußnahme auf Entwurfsentscheidungen ermöglicht werden und andererseits das Verständnis der Entwickler dafür gefördert werden, ob und welche DV-Leistungen für eine angemessene Unterstützung der Arbeitsaufgaben zur Verfügung gestellt werden sollen. Dabei ist zu berücksichtigen, daß Entwurfsentscheidungen, die für die Einpassung des DV-Systems in die Arbeitsaufgaben relevant sind, explizit vollzogen werden und daß die Kommunikation mit verschiedenen Personen und Personengruppen — ebenso wie ihre möglicherweise differierenden Sichten des zu entwickelnden Systems — dokumentiert werden. Für die Entwickler ist es darüber hinaus wichtig, daß die zu erstellenden Modelle und Dokumente die Entwicklungs- und Implementierungsarbeit erleichtern und nicht nur einen für ihre weitere Arbeit unbrauchbaren zusätzlichen Dokumentationsaufwand bedeuten.

Verfahren und Techniken zur Anforderungsermittlung müssen daher sowohl aufgabenbezogen, kommunikationsunterstützend und verständnisfördernd, als auch im Hinblick auf die weiteren Entwicklungsschritte effizient nutzbar sein. Was jedoch diese Kriterien im einzelnen bedeuten, läßt sich nur aufgrund der Analyse praktisch durchgeführter Entwicklungsprojekte feststellen und bewerten.

2.2 Anforderungsermittlung mit Aufgabennetzen

Der Anwendungsbereich beschränkt sich auf die Einbettung interaktiver Systeme in das menschliche Arbeitshandeln. Bezüglich der Modellierung werden dabei nur die Arbeitsprozesse erfaßt, die sich auf objektmanipulierende Tätigkeiten zurückführen lassen.

Die hier skizzierte Vorgehensweise ist insofern idealtypisch, als sie nicht die Vielfalt der situationsbedingten und personenabhängigen Verfahrensweisen in praktischen Projekten berücksichtigt. Außerdem gibt sie lediglich die Sicht der Entwickler wieder.

Diese müssen sich zuerst einmal ein Bild von der Organisation machen. Sie müssen sich insbesondere vertraut machen mit:

- der Organisationsstruktur und den Arbeitsplätzen,
- den Zuständigkeiten und Entscheidungsbefugnissen,

- den Konventionen für den Gebrauch von Formularen und für die Erstellung interner Berichte,
- der jeweils verwendeten Fachsprache, die je nach Abteilung unterschiedlich sein kann und
- den verwendeten technischen Konzepten und Entwicklungsstrategien.

Um einen ersten Überblick zu gewinnen, sollte mit den an der Systementwicklung beteiligten und den von ihr betroffenen Personen eine gemeinsame **Projektsprache** entwickelt werden. Dies kann in Form von Gesprächen, Interviews, Beobachtungen oder auch anhand von Arbeitsplatzbeschreibungen geschehen. In der Regel ist es eine Mischung aus all diesen Möglichkeiten.

Die Begriffe dieser Projektsprache müssen einen Bezug zur Benutzerfachsprache aufweisen und für alle an der Entwicklung Beteiligten verbindlich sein. In einem weiteren Schritt sollte die Projektsprache bereinigt werden, d.h. alle Synonyme und Homonyme sollten entfernt werden. Bei größeren Organisationen oder komplexeren Problemstellungen empfiehlt es sich, ein alphabetisches Verzeichnis dieser Begriffe (Glossar) anzulegen; es erleichtert das Verständnis und dient als Bezugspunkt in den weiteren Entwicklungsschritten.

Auf der Grundlage dieser Terminologie können dann die Arbeitsplätze genauer untersucht werden. Die Anforderungsermittlung kann dabei grob in zwei Schritte unterteilt werden: die **Aufgabenanalyse**, häufig auch als *IST-Konzept* bezeichnet und die Beschreibung der **Rechnereinbettung**, d.h. die Erstellung eines *SOLL-Konzeptes*.

Ausgangspunkt ist das Konzept der **funktionellen Rolle**. Eine funktionelle Rolle ist über die Aufgaben definiert, die von einer Person oder Personengruppe wahrgenommen werden.² Dieses Konzept dient dazu, sowohl die bestehenden als auch die mit der Einführung des DV-Systems neu zu definierenden Arbeitsaufgaben mit den Zielen der Organisation und den Interessen und Wünschen der Benutzer nach einem zufriedenstellenden und qualifizierenden Arbeitsplatz in Verbindung zu bringen.

Bei der Aufgabenanalyse werden nur diejenigen Arbeitsprozesse modelliert, die sich auf objektmanipulierende Tätigkeiten zurückführen lassen. Aktivitäten wie z.B. Beratung, Überwachung und Planung lassen sich nicht sinnvoll auf solche Tätigkeiten zurückführen. Ineinandergreifende Folgen von einzelnen **Tätigkeiten**, die dazu dienen, den Bearbeitungszustand von **Objekten** zu verändern, werden als **Arbeitsabläufe** bezeichnet. Als Objekte werden alle Gegenstände betrachtet, die im Rahmen der Arbeitsprozesse erzeugt, verändert, archiviert oder weitergegeben werden. Dies sind z.B. Rechnungen, Aufträge, Zahlungen, Karteikarten oder Bücher. Wünsche und Vorstellungen sowie Ideen oder abstrakte Begriffe werden im Sinne der Aufgabennetze nicht als Objekte behandelt. Objekte können gesammelt, zusammengefaßt oder abgelegt werden. Dies geschieht in sogenannten **Archiven**. Beispiele hierfür sind Karteien und Ordner aber auch Verzeichnisse und Listen.

Nach der Aufgabenanalyse wird festgestellt, ob und in welchem Umfang Tätigkeiten automatisiert werden sollen und in welchem Umfang neue, durch die Einführung eines DV-Systems möglich gewordene Tätigkeiten, hinzukommen. Bei der Rechnereinbettung werden die Systemfunktionen und die Objekte modelliert, die jeweils am Bildschirm eingegeben oder angezeigt bzw. ausgedruckt werden sollen. Hinzu kommen die veränderten Arbeitsabläufe. Die Modellierung geht dabei nur soweit wie es für das Verständnis der Einbettung der Systemfunktionen in das Arbeitshandeln notwendig ist. Für eine detaillierte Spezifikation der Benutzungsschnittstelle sind eigene Konzepte und Techniken erforderlich.³

Funktionelle Rollen, Arbeitsabläufe, Tätigkeiten, Objekte und Archive werden durch **Aufgabennetze** beschrieben. Als halbformales Darstellungsmittel wird eine Variation von Petri-Netzen benutzt, deren Grundinterpretation mit Kanal-Instanzen-Netzen vergleichbar ist.⁴ Wichtig ist,

²Ein differenzierteres Konzept wird in Oberquelle (1987b) beschrieben.

³Vgl. Keil-Slawik, Pasch (1987)

⁴Vgl. Richter (1983), Reisig (1985) sowie Oberquelle (1987b)

daß alle Objekte und Funktionen mit entsprechenden Erläuterungen in **Lexika** verzeichnet werden. Ein Eintrag in einem Objektlexikon enthält neben dem Namen des Objektes eine kurze Objektbeschreibung, eine Liste sämtlicher Bearbeitungszustände (Attribute) sowie Strukturmerkmale wie z.B. Typ und Wertebereich. Im Funktionslexikon werden unter dem Funktionsbezeichner eine Kurzbeschreibung, Hinweise zur Handhabung sowie die erwartete Häufigkeit der Benutzung angegeben.

An dieser Stelle ist es wichtig darauf hinzuweisen, daß das Ausdrucksrepertoire der Aufgabennetze beschränkt ist und nicht alle für das menschliche Arbeitshandeln wesentlichen Aspekte modelliert werden können. Arbeitsabläufe werden letztendlich auf Bearbeitungszustände von Objekten zurückgeführt. Es werden also nicht die tatsächlichen Handlungen des Menschen modelliert, sondern nur, welche Zustandsveränderungen er mit seinen Tätigkeiten an den Objekten bewirkt. Auf dieser Ebene besteht auch der Anknüpfungspunkt zu Funktionen und Zuständen eines DV-Systems.

Für eine arbeitswissenschaftliche Analyse bieten solche Modelle zwar Anknüpfungspunkte aber keine ausreichende Grundlage. Sie können die Aufgabengestaltung nach arbeitswissenschaftlichen Kriterien nicht ersetzen, wohl aber helfen, das Ergebnis zu dokumentieren.

Desweiteren ist darauf zu achten, daß räumliche Zuordnungen, Zeitbedingungen, sowie Entscheidungsbefugnisse und Ermessensspielräume, soweit sie für die Systementwicklung wichtig sind, gesondert dokumentiert werden müssen.

3 Ausgangspunkt der Bewertung

3.1 Einordnung in den Modellierungsprozeß

Traditionell steht auch bei der Entwicklung von Methoden und Techniken zur Gestaltung interaktiver Systeme eine produktbezogene Sichtweise im Vordergrund. Dies äußert sich in zweierlei Hinsicht:

- Bei der Konzeption stehen die formalen Eigenschaften im Vordergrund der Betrachtung. Entscheidend ist der **Modellierungsgegenstand** und nicht der **Modellierungsprozeß**. Überlegungen zur Handhabbarkeit, individuellen Anpaßbarkeit und zur Eignung in unterschiedlichen Projektsituationen und für unterschiedliche Anwendungen haben in der Regel keinen Einfluß auf die konzeptionelle Gestaltung.⁵ Anwendungsbeispiele werden in der Regel erst nach Abschluß der Konzeptionsphase zur didaktischen Unterstützung ausgearbeitet.
- Methoden und Konzepte werden als weitgehend unveränderbar betrachtet. Das bezieht sich sowohl auf die individuelle Anpassung im Rahmen eines Entwicklungsprojektes als auch auf das Ausprägen von Stilen und Dialekten durch verschiedene Anwenderorganisationen. Insbesondere die Notwendigkeit zur Anpassung an neue Situationen und Bedürfnisse des Methodenanwenders wird nicht angemessen gewürdigt.⁶

Im Gegensatz zu einer rein produktorientierten Sichtweise liegt der Konzeption der Aufgabennetze eine primär prozeßorientierte Strategie zugrunde, d.h. im Vordergrund steht der Modellierungsprozeß und seine Einbettung in die Systementwicklung.

Anhand einer Fallstudie⁷, die im Rahmen eines Methodenvergleichs durchgeführt wird, zeigt sich die prinzipielle Eignung von (Petri-)Netzen für die Systementwicklung. Daran anknüpfend wird ein Verfahren zur aufgabenbezogenen Anforderungsermittlung mit Aufgabennetzen ausgearbeitet.⁸ Das Ziel ist, ein praktikables Verfahren für die Gestaltung interaktiver Systeme

⁵Vgl. dazu die Stellungnahmen zu verschiedenen Modellierungsansätzen in Wedde (1983)

⁶Vgl. Floyd, Pasch (1985)

⁷Kämper, Schneider (1983)

⁸Vgl. Keil-Slawik (1985)

zu entwickeln, das die Defizite traditioneller Entwicklungsmethoden, deren Schwerpunkte im Bereich der Funktionalisierung und Algorithmisierung liegen⁹, ausgleicht. Anhand des praktischen Einsatzes werden die Relevanz und die Brauchbarkeit der Konzepte und der zugrundegelegten Annahmen überprüft. Dieses Vorgehen fördert auf der einen Seite das Verständnis für die bei der Systementwicklung ablaufenden Prozesse und hilft auf der anderen Seite, die dabei verwendeten Verfahren und Hilfsmittel zu verbessern und anzupassen. Die grundlegenden Konzepte sind dabei unverändert geblieben.

Bei der Auswertung geht es folglich nicht darum aufzuzeigen, inwieweit Aufgabennetze besser oder schlechter sind als andere Techniken; dazu fehlt die Vergleichsgrundlage. Vielmehr sollen die zugrundegelegten Annahmen kritisch überprüft werden, um festzustellen, was sich bewährt hat und was nicht.

3.2 Rahmenbedingungen und Bewertungsgrundlage

Bei dieser Auswertung werden insgesamt vier Projekte und drei Fallstudien berücksichtigt. Bei den Fallstudien handelte es sich um Problemstudien, bei denen in Gesprächen mit potentiellen Benutzern aus verschiedenen Organisationen für einen Problembereich eine Anforderungsdefinition erstellt worden ist. Die Problembereiche sind hier: die Ausleihverwaltung von Büchern, die Vermittlung von Mitfahrern sowie die computergestützte Projektierung der Fertigung elektronischer Bauteile.

In den Projekten ist jeweils ein interaktives System entwickelt worden, das auch unter den gegebenen industriellen bzw. verwaltungstechnischen Rahmenbedingungen eingesetzt wird. Dabei handelt es sich um ein Material- und Lagerverwaltungssystem¹⁰, ein Zeitschriftenersfassungs- und -umlaufsystem¹¹, ein System zur gewerblichen Anzeigenverwaltung¹² sowie um ein System zur Unterstützung der Erfassung und Auswertung von Tarifverträgen.¹³

Kennzeichnend für die Projekte ist, daß sie vergleichsweise kleine Anwendungen darstellen; mit der Entwicklung sind in der Regel zwei bis vier Personen über einen Zeitraum von 1—1½ Jahren beschäftigt gewesen. Außergewöhnlich ist, daß alle Projekte in einem äußerst partizipativen Klima abgewickelt werden konnten, bei denen die späteren Benutzer — meist zwischen drei und sechs Personen — zum Teil weitreichende Entscheidungsmöglichkeiten hatten und demzufolge hoch motiviert waren. Durch die kleinen Gruppen war für die Projektabwicklung ein hohes Maß an Flexibilität gegeben. Diese Flexibilität ist aber durch 'harte' Vorgaben bezüglich der Anforderungen an das System teilweise eingeschränkt worden.

Zur Auswertung dienen Projektunterlagen, Diplomarbeiten und Dissertationen, die im Zusammenhang mit diesen Projekten und Fallstudien angefertigt worden sind. Daneben haben mit fast allen Entwicklern Gespräche statt gefunden, sowie in zwei Fällen auch mit den Benutzern. Weitere Erkenntnisse haben sich aus der Aufbereitung einzelner Materialien für Lehrveranstaltungen und Projekte ergeben, die im Rahmen des Softwaretechnik-Curriculums an der TU Berlin durchgeführt worden sind.

4 Erfahrungen im praktischen Einsatz

In diesem Abschnitt geht es in erster Linie um den Stellenwert der Aufgabennetze. Weitere, für die Entwicklung interaktiver Systeme wesentliche Aspekte wie z.B. Projektmodelle oder Zeitbeschränkungen werden nicht behandelt, obwohl sie auf die Bewertung der hier angesprochenen Punkte nicht ohne Einfluß sind. Die meisten der in den verschiedenen Abschnitten aufgeführten

⁹Vgl. Floyd (1984)

¹⁰Holland (1989), Zumkley (1988)

¹¹Staeck, Vanselow (1987)

¹²Lange, Ruholl (1986), Vogel (1986)

¹³Ansorge, Köther, Wicke (1986), Floyd, Mehl, Reisin, Schmidt, Wolf (1989)

Aspekte und Begründungen weisen starke wechselseitige Abhängigkeiten untereinander auf und können erst im Rahmen einer ganzheitlichen Betrachtung ihr eigentliches Gewicht entfalten. Was jeweils angemessen ist, hängt also von der Projektkonstellation ab und kann nicht aufgrund eines allgemein gültigen Kriterienkataloges entschieden werden.

4.1 Verständnisförderung

In allen Berichten der Entwickler wird betont, daß Aufgabennetze in den jeweiligen Projekten eine gute Unterstützung boten, um sich ein Verständnis der Arbeitsaufgaben zu erarbeiten. So konnten in einem Fall sogar inkonsistente Bearbeitungsmodalitäten bei der Rabattberechnung aufgezeigt werden. Sie bedeuteten für das entsprechende Unternehmen jeden Monat erhebliche finanzielle Verluste. Allerdings muß hier betont werden, daß dies sicherlich auch mit anderen Instrumenten wie z.B. SADT gelungen wäre.

Es hat sich allerdings gezeigt, daß Aufgabennetze allein nicht ausreichend sind. Organisationen haben keine a priori gegebene Realität, die lediglich zu beobachten ist und dann beschrieben werden kann.¹⁴ Für die Entwickler zeigt sich dies u.a. darin, daß sie herausfinden müssen, wer ihnen überhaupt verbindliche Angaben machen kann und wie sie ihre Vorgehensweise und auch Modellierungskonzepte anpassen müssen, um die in der jeweiligen Projektsituation relevanten Aspekte effektiv bearbeiten zu können:

„Für uns stellte sich erst mit der Zeit heraus, welcher von unseren Gesprächspartnern für welche Themenbereiche entscheidungsbefugt war.

Wir versuchten alles, was an Anforderungen an uns herangetragen wurde, in das System zu integrieren. Dabei gingen wir von der Annahme aus, daß die Anwender ja wüßten, welchen Anforderungen das System genügen sollte. Erst mit der Zeit merkten wir, daß die formulierten Anforderungen oft widersprüchlich und ungenau waren und das System zu einer nicht mehr beherrschbaren Komplexität anwachsen ließen.“¹⁵

Doch auch im günstigsten Fall muß sich der Entwickler immer wieder entscheiden, welche Aspekte er jeweils modellieren bzw. dokumentieren will. Bei vielen Arbeitsabläufen gibt es beispielsweise eine Fülle von Ausführungsmöglichkeiten, Unterbrechungen (z.B. durch Telefonanrufe) und Sonderfällen, die nicht alle gleichermaßen erkannt und berücksichtigt werden können. Zudem sind sie personen-, situations- und zeitbedingt unterschiedlich. Aus diesen Gründen können Aufgabennetze auch nicht das einzige Beschreibungsmittel sein; sie müssen durch geeignete Zusatzbeschreibungen und Kommentierungen ergänzt werden.¹⁶

Neben solchen Zusatzbeschreibungen müssen die Entwickler meist auch eine geeignete Pragmatik für die Aufgabenbeschreibung entwickeln. Beispielsweise gibt es in Organisationen keine funktionellen Rollen. Dieses Konzept ist ein Hilfsmittel, um sinnvolle Unterscheidungen treffen zu können. Je nach Situation kann z.B. zur Charakterisierung einer funktionellen Rolle die Stellung im Betrieb (z.B. Leiter), die Qualifikation (z.B. Sachbearbeiter) oder es können auch unterschiedliche soziale oder technische Arbeitsmilieus in denen ein und dieselbe Person arbeitet (z.B. Empfang, Labor, Sekretariat etc.), benutzt werden. Dabei sind noch entsprechende Vertretungsregeln auch für Teilbereiche zu berücksichtigen.

FAZIT: Das Ergebnis einer Aufgabenanalyse ist nicht eine objektive Beschreibung der Realität, sondern eine Realitätskonstruktion, die das Verständnis der Entwickler in Abstimmung mit den Benutzern widerspiegelt. Modellierungskonzepte wie Rollen und Aufgaben haben keine allgemeingültige präzise Definition. Nicht die Anwendung formaler Kriterien schafft das Verständnis für den Anwendungsbereich, sondern die sinnvolle Begründung der mit Hilfe dieser Konzepte getroffenen Unterscheidungen.

¹⁴Vgl. Weick (1985) und Westerlund, Sjöstrand (1981)

¹⁵Lange, Ruholl (1986)

¹⁶Vgl. Ansorge, Köther, Wicke (1986); S. 16

4.2 Entwickler–Benutzer–Kommunikation

Obwohl bereits bei der Konzeption der Aufgabennetze die Prozesse ihrer Erstellung und Verwendung im Vordergrund standen, ist dem vorgehend angesprochenen konstruktiven Aspekt noch zu wenig Beachtung geschenkt worden.

Die Erarbeitung einer gemeinsamen Projektsprache mit den Benutzern vor der eigentlichen Aufgabenmodellierung (vgl. Abschnitt 2.2) wurde zuerst von Jürgen Pasch bei der Entwicklung des Anzeigenverwaltungssystems explizit eingeführt und dokumentiert¹⁷; es gehört mittlerweile zum Repertoire der aufgabenbezogenen Anforderungsermittlung.

Im Gegensatz zur Festlegung der Fachterminologie ist die Erfahrung mit den Aufgabennetzen zur Unterstützung der Entwickler–Benutzer–Kommunikation sehr widersprüchlich. Zwar wird generell die Einfachheit und Überschaubarkeit der Netze von den Entwicklern herausgestellt, doch ist die Verständigung mit den Benutzern anhand von Aufgabennetzen auf gemeinsame Treffen beschränkt geblieben, in denen die Entwickler versucht haben, mit Hilfe der Netze ihr Verständnis der Arbeitsaufgaben zu überprüfen. In keinem Projekt haben Benutzer selbst Netzmodelle oder Teile davon erstellt.

Häufig sehen Benutzer nicht ein, warum sie sich auf die Beschreibungssprache der Entwickler einlassen müssen. Dies hat unterschiedliche Gründe:

„Das Lesen der Netze war einigen Mitarbeitern zu umständlich, da es ihnen oft schwer fiel, die Arbeitsabläufe wiederzufinden, mit denen sie sich Tag für Tag auseinandersetzen“.¹⁸

Ein Teil der widersprüchlichen Aussagen wird verständlich, wenn man sich den Unterschied im Verständnis von Netzmodellen, die den eigenen Arbeitsbereich betreffen und an deren Entwicklung man selbst beteiligt war, gegenüber 'fremden' Modellen vor Augen hält:

„Schwierigkeiten ergaben sich nur bei einem Wechsel zu Netzen anderer Rollen, wobei nur ein Teilaspekt oder eine Teilaufgabe zu betrachten war. Durch den hohen Detaillierungsgrad der Aufgabennetze, und das dadurch entstandene Aufteilen in Verfeinerungsstufen war es dann den meisten Gesprächspartnern nicht möglich, einen Überblick zu gewinnen“.¹⁹

Weitere Schwierigkeiten tauchen bei der Modellierung der Rechnereinbettung auf:

„Um die Möglichkeiten des Einsatzes und der Gestaltung von Softwaresystemen aufzuzeigen, bietet es sich bei ungeschulten Benutzern an, im Vorfeld die Handhabung mit Hilfe einfacher Systeme deutlich zu machen. Hingegen halten wir es nicht für sinnvoll, Darstellungsmittel wie Interaktions- und Aufgabennetze einzusetzen, solange sie nicht vom Benutzer zur Erledigung eigener Arbeiten adaptierbar sind“.²⁰

Die Schwierigkeit, Netze zu lesen und zu interpretieren, zeigte sich auch bei der Aufbereitung von Fallstudien für Lehrveranstaltungsprojekte. Ohne Tutor, der ihnen entsprechende Fragen beantwortete, war es den Studenten nicht möglich, selbst bei einer umfangreichen und detaillierten Netzbeschreibung sich ein adäquates Bild von den modellierten Arbeitsabläufen zu machen. Überraschenderweise waren zu Beginn die Lexika zum Verständnis teilweise wichtiger als die Netze selbst, da sie mehr Informationen über die bearbeiteten Objekte in gedrängter Form geben als die graphische Darstellung, die in der Regel mit sehr kurz gehaltenen Beschriftungen auskommen muß.

FAZIT: Modellierungs- und Darstellungstechniken sind in erster Linie ein Instrument der Entwickler. Benutzer haben mehr Interesse an der Vorstellung ihrer Probleme und Wünsche als an einer Modellierung ihrer Tätigkeiten. Darstellungen und Modelle, die nicht an sinnlich erlebte Erfahrungen anknüpfen, werden nicht verstanden.

¹⁷Pasch, Vogel (1985); S. 4–15

¹⁸Holland (1989); S. 67

¹⁹Zumkley (1988); S. 71

²⁰Staeck, Vanselow (1987); S. 122

4.3 Handhabbarkeit

Graphische Darstellungen sind naturgemäß sehr aufwendig zu erstellen und schwer zu ändern. Für 'Reinzeichnungen' benötigen Entwickler selbst mit Rechnerunterstützung ein Vielfaches der Zeit, die sie zur Erstellung von Handskizzen brauchen. Die übersichtliche und überschneidungsfreie Anordnung der graphischen Elemente sowie ihre Beschriftung erfordern häufig mehrere Versuche, bis das Ergebnis zufriedenstellend ist.

Obwohl Aufgabennetze mittlerweile nur noch sieben unterschiedliche graphische Elemente beinhalten (fünf Knotentypen und zwei Arten von Kanten), stöhnen alle Entwickler über den hohen Erstellungsaufwand. In einem Projekt beispielsweise, in dem von einer anderen Organisation bereits eine dokumentierte Vorstudie mit Hilfe von Aufgabennetzen ausgearbeitet worden war, benutzten die Entwickler, obwohl sie auf der Vorstudie aufsetzten und mit der Technik der Aufgabennetze bestens vertraut waren, für Aufgabenskizzen eine reduzierte Form von SADT ein. Der Grund liegt im erheblich geringeren Zeichenaufwand, denn es gibt nur noch je einen Knoten- und Kantentyp. Außerdem ist die Erstellung und Dokumentation eines vollständigen Modells als nicht notwendig erachtet worden; zur Verständnisbildung haben die Skizzen ausgereicht.

Eine weitere Erfahrung der Entwickler ist, daß sich der Anteil an Beschriftungen, die keinen zusätzlichen Informationsgehalt aufweisen, mit der Anzahl der unterschiedlichen graphischen Elemente schnell vergrößert. Ein typisches Beispiel für solche in Netzen häufig auftretende Mehrfachbeschriftungen ist: „Schriftstück ablegen“ als Tätigkeit und „SCHRIFTSTÜCK abgelegt“ als Beschreibung des erreichten Bearbeitungszustandes. Modellierungskonzepte, wie z.B. die RFA-Netze²¹, die über zwanzig verschiedene graphische Elemente enthalten, können bei Benutzern wie auch Entwicklern aus diesem Grund schnell auf Akzeptanzprobleme stoßen.

Als Defizit der Aufgabennetze wird die Tatsache gewertet, daß keine ausreichende Anbindung an formale Konzepte im Hinblick auf die technische Realisierung besteht.²² Zusätzlich wird betont, „...daß eine Datenmodellierung parallel zur Problemanalyse erfolgen muß“²³, da Typen und Relationen gewissermaßen bereits bei der Objektmodellierung 'abfallen'. Dabei ist aber zu berücksichtigen, daß je nach technischem Entwicklungsmilieu und der Zielsetzung (z.B. die Realisierung einer Benutzungsschnittstelle als Prototyp) unterschiedliche Formalismen erforderlich sind. Die ausgeprägte Objektorientierung kann dabei noch zusätzliche Schwierigkeiten bereiten. Bei der Gestaltung der Benutzungsschnittstelle tritt z.B. das Problem auf, daß durch die Technik der interreferentiellen Ein-/Ausgabe²⁴ unklar ist, welche Objekte tatsächlich über die Schnittstelle 'wandern'.

FAZIT: Modellierungs- und Darstellungstechniken müssen die bei der Entwicklung auftretenden Arbeitsprozesse angemessen unterstützen. Zusätzlicher Dokumentations- und Zeichenaufwand ist nur gerechtfertigt, wenn dadurch die Arbeit in späteren Entwicklungsschritten wesentlich erleichtert wird, z.B. durch die Generierung von später benötigten Dokumenten oder Programmteilen. Bezüglich der Einbindung dieser Techniken in die Software-Entwicklung gibt es insbesondere im Hinblick auf neuere Konzepte wie Hypertextsysteme, objektorientierte Programmierung und graphische Benutzungsschnittstellen erhebliche Defizite.

4.4 Stellenwert von Dokumenten

Obwohl die untersuchten Projekte nicht repräsentativ sind, zeigen sie doch grundsätzliche Probleme auf. Bei der Befragung der Entwickler hat sich u.a. ergeben, daß die vollständige Dokumentation der Aufgabenanalyse in der Regel erst nach der Realisierung des Systems erfolgt ist. Dies deckt sich auch mit der Beobachtung von Jan Witt, der in Bezug auf qualitätssichernde Maßnahmen bei der Software-Entwicklung feststellt:

²¹Vgl. Oberquelle (1987a)

²²Vgl. Oberquelle (1987b); S. 93

²³Vogel (1986); S. 29

²⁴Inter-Referential I/O, Draper (1986)

„In der Softwarewelt gilt leider oder vielleicht aus grundsätzlichen Gründen immer noch, daß zuerst das Gewünschte in der Rohform erstellt wird und dann erst dem Besucher von außen Zutritt gestattet wird, wenn das Schlimmste schon weggeputzt ist“.²⁵

Hinzu kommt, daß die Anfangsphasen der Systementwicklung als Orientierungs- und Etablierungsphase charakterisiert werden können²⁶, in denen Vorstellungen, Annahmen, Wünsche und Anforderungen noch häufig und zum Teil grundlegend revidiert werden. Die Motivation, unter solchen Rahmenbedingungen einen hohen Aufwand in die Ausgestaltung von Dokumenten und formalen Verfahren zu stecken, ist insofern nicht sehr hoch.

Wenn aber erst einmal erkannt und vereinbart worden ist, welche Systemleistungen zur Verfügung gestellt werden sollen, wird meist keine Notwendigkeit mehr gesehen, alle Netze vollständig und in gut lesbarer Form zu dokumentieren. Diese Tatsache wird auch verständlich, wenn man berücksichtigt, daß in den nachfolgenden Entwicklungszyklen Änderungen und Erweiterungen in der Regel am bereits existierenden System vollzogen werden, d.h. auf der Grundlage des Datenmodells, der Systemfunktionen oder der Benutzungsschnittstelle. Arbeitsschritte, die lediglich darin bestehen, ein bestehendes Dokument in ein anderes zu überführen (transformieren), ohne daß dabei wesentliche neue Erkenntnisse abfallen, werden, soweit sie nicht aus projekttechnischen Gründen erforderlich sind, weitestgehend vermieden.

Zwar eignen sich Aufgabennetze hervorragend, um beispielhaft Arbeitsabläufe zu skizzieren, doch führt der Anspruch an ein vollständiges Modell des Anwendungsbereiches schnell dazu, daß Dokumente aufwendig und unhandlich werden. Allein die Netzbeschreibungen für einen Teil der gewerblichen Anzeigenverwaltung umfassen zusammen mit den Lexika aber ohne die erforderliche Zusatzdokumentation bereits über 90 Seiten.

Da ohnehin ein Großteil der für die Benutzung eines interaktiven Systems wesentlichen Aspekte nicht über beschreibende Dokumente erfahrbar und vermittelbar ist, geht man mittlerweile dazu über, Prototypen zu entwickeln. Der Wert eines Prototypen hängt aber maßgeblich von seiner systematischen Auswertung ab²⁷; Aufgabennetze können hier zur Spezifikation der Auswertungsgrundlage eingesetzt werden.

FAZIT: Wesentlich zum Verständnis dessen, was interaktive Systeme in einem gegebenen Organisationskontext leisten sollen, sind nicht so sehr die Dokumente, sondern die Prozesse, die zu ihrer Entstehung führen. Der Sinngehalt von Modellen und Begriffen wird durch die an der Entwicklung beteiligten Menschen konstruiert. Modellierungs- und Darstellungstechniken müssen diese Konstruktionsprozesse angemessen unterstützen. Je weniger dabei der Modellierungsgegenstand bereits formal durchstrukturiert ist, desto geringer ist auch der Wert von Formalismen.

5 Zusammenfassung

Aufgabennetze erfüllen die Funktion, Aspekte aus der Benutzersicht zu beschreiben und sie mit der vom Hersteller entwickelten Sicht zu verbinden. Das beinhaltet sowohl eine Öffnung der Softwaretechnik für Ansätze und Verfahren, die auf die Gestaltung persönlichkeitsförderlicher Arbeitsplätze zielen, als auch eine effektive und praktisch handhabbare Unterstützung des weiteren Entwicklungsprozesses.

Insgesamt ist aber festzustellen, daß der Wert von Methoden wie der aufgabenbezogenen Anforderungsermittlung im Rahmen der Systementwicklung beschränkt ist. Viele Probleme und relevante Aspekte der Systemgestaltung werden durch Methoden und Darstellungstechniken nicht oder nur unzureichend unterstützt. Oft haben ausführliche Gespräche, Messevorführungen, Prototypen, Bilder, Mock-Up's und Szenarien einen erheblich größeren Stellenwert als ausführliche, auch graphisch aufbereitete Dokumente.

²⁵ zitiert nach Hesse (1988)

²⁶ Vgl. Reisin, Schmidt (1988); S.46

²⁷ Siehe Floyd (1984); S. 5

Der Wert für den Softwareentwickler liegt weniger darin, vollständige, eindeutige und konsistente Dokumente erstellen zu können. Vielmehr haben Aufgabennetze den Charakter von Krücken oder Hilfsrädern: sie stützen solange man sich im Anwendungsfeld noch nicht richtig bewegen kann. Sobald man aber problemlos gehen bzw. fahren kann, werden sie nicht mehr benötigt. Sie mitzuschleppen, wäre unnötiger Ballast.

Bei der Bewertung von Modellierungs- und Darstellungstechniken geht es nicht um eine Entscheidung für oder gegen formale oder informelle Darstellungen bzw. vollständige Dokumente oder die prozeßbezogene Kommunikation. Vielmehr geht es um das Problem, zu welchem Zeitpunkt und in welcher Situation welcher Grad an formaler Rigidität wünschenswert oder notwendig ist.

Die vorliegende Auswertung weist darauf hin, daß diese Frage weder aus immanenten Eigenschaften von Formalismen und mathematischen Theorien noch aus dem Modellierungsgegenstand heraus allein beantwortet werden kann. Zwar gelten die vorliegenden Befunde nur für einen relativ kleinen Ausschnitt der Projektkonstellationen, die bei der Entwicklung interaktiver Systeme auftreten, doch machen sie deutlich, daß in sich geschlossene, konsistente und eindeutige Formalismen nicht notwendigerweise mit der Forderung nach angemessener Unterstützung des menschlichen Handelns einhergehen.

Sowohl für die Entwicklung als auch für die Benutzung interaktiver Systeme gilt, daß die angemessene Unterstützung des menschlichen Arbeitshandelns letztlich das entscheidende Qualitätskriterium darstellt. Dazu ist es aber notwendig, den Menschen mit seinen Fähigkeiten und Bedürfnissen in den Vordergrund zu stellen und nicht die Methoden, Formalismen und Maschinen.

Literatur

- Ansorge, P., Köther, B., Wicke, W.: **Untersuchung der Möglichkeiten und Probleme des EDV-Einsatzes im WSI-Tarifarchiv**. BAIT, Beratungs- und Forschungsinstitut Arbeit und Informationstechnologie, Dortmund, September 1986
- Draper, S.W.: **Display Managers as the Basis for User-Machine Communication**. In: Norman, D.A., Draper, S.W. (Eds.): *User Centered System Design. New perspectives on Human-Computer Interaction*. Lawrence Earlbaum: Hillsdale London, 1986
- Floyd, C.: **A Systematic Look at Prototyping**. In: Budde, R., Kuhlenkamp, K., Mathiassen, L., Züllig-hoven, H.: *Approaches to Prototyping*. Springer: Berlin Heidelberg New York Tokyo, 1984
- Floyd, C.: **STEPS — Eine Orientierung der Softwaretechnik auf sozialverträgliche Technikgestaltung**. In: Riedemann, E., Hagen, U.v., Heß, K.—D., Wicke, W. (Hrsg.): *10 Jahre Informatik und Gesellschaft — Eine Herausforderung bleibt bestehen*. Forschungsbericht Nr. 227; Universität Dortmund, 1986. Auch abgedruckt in: *Informatik Forum*; Band 2, Heft 2, Juli 1987
- Floyd, C., Pasch, J.: **Methoden für den Entwurf großer Softwaresysteme**. In: Morgenbrod, H., Remmele, W. (Hg.): *Entwurf großer Softwaresysteme*. Teubner: Stuttgart, 1985
- Floyd, C., Mehl, M., Reisin, F.-M., Schmidt, G., Wolf, G.: **Partizipative Entwicklung transparenz-schaffender Softwaresysteme für DV-gestützte Arbeitsplätze**. Abschlußbericht F&E Projekt; erscheint im Rahmen der Publikationsreihe „Mensch und Technik: Sozialverträgliche Technikgestaltung“ der Landesregierung von Nordrhein-Westfalen
- Hesse, W.: **Verfahren zur Software-Qualitätssicherung**. Editorial; *Informatik-Spektrum*; Band 11, Heft 6, Dezember 1988)
- Holland, C.: **Analyse und Spezifikation einer Auftragsverwaltung und Materialplanung anhand von Aufgabennetzen**. Studienarbeit; TU Berlin, Januar 1989

- Kämper, S., Schneider, N.: **Beschreibung verschiedener Petri-Netz-Interpretationen zur Modellierung soziotechnischer Systeme.** Diplomarbeit; TU Berlin, Oktober 1983
- Keil-Slawik, R.: **KOSMOS — Ein Konstruktionsschema zur Modellierung offener Systeme als Hilfsmittel für eine ökologisch orientierte Softwaretechnik.** Dissertation; TU Berlin, März 1985
- Keil-Slawik, R., Pasch, J.: **Methodisches Vorgehen bei der Entwicklung von Dialogschnittstellen.** 8. Int. Kongress Datenverarbeitung im europäischen Raum; ADV; Wien, 30.3. — 3.4.1987
- Lange, W., Ruholl, H.: **Erfahrung mit Prototyping bei der Zweiten Hand Verlags GmbH.** Diplomarbeit; TU Berlin, August 1986
- Oberquelle, H.: **Benutzerorientierte Beschreibung von interaktiven Systemen mit RFA-Netzen.** In Schönplüg, W., Wittstock, M. (Hg.): *Software-Ergonomie '87.* Teubner: Stuttgart, 1987(a)
- Oberquelle, H.: **Sprachkonzepte für benutzergerechte Modelle.** Springer: Berlin Heidelberg New York, 1987(b)
- Pasch, J., Vogel, N.: **Anforderungsdefinition für eine gewerbliche Anzeigenverwaltung.** Manuskript; TU Berlin, Sommer 1985
- Reisig, W.: **Systementwurf mit Netzen.** Springer: Berlin Heidelberg New York Tokyo, 1985
- Reisin, F.-M., Schmidt, G.: **STEPS — Evolutionär und Partizipativ.** Computer Magazin; Heft 7/6, 1988
- Richter, G.: **Realitätsgetreues Modellieren und modellgetreues Realisieren von Bürogesehen.** In: Wißkirchen et al (Hg.): *Informationstechnik und Bürosysteme.* Teubner: Stuttgart, 1983
- Staeck, R., Vanselow, R.: **Prototypenorientierte Entwicklung eines Softwaresystems zur Zeitschrifteneingangs- und -umlaufkontrolle unter besonderer Berücksichtigung der Arbeitsabläufe der Mitarbeiter(innen) der zentralen Dokumentation der Firma Schering Berlin.** Diplomarbeit; TU Berlin, Oktober 1987
- Weick, K.E.: **Der Prozeß des Organisierens.** Suhrkamp: Frankfurt, 1985
- Wedde, H. (Ed.): **Adequate Modeling of Systems.** Springer: Berlin Heidelberg New York, 1983
- Westerlund, G., Sjöstrand, S.-E.: **Organisationsmythen.** Klett-Cotta: Stuttgart, 1975
- Vogel, N.: **Untersuchung von einigen Methoden zur Konstruktion von semantischen Datenmodellen im Hinblick auf die Verträglichkeit mit Aufgabennetzen.** Diplomarbeit; TU Berlin, Mai 1986
- Zumkley, M.: **Analyse und Spezifikation einer Materialverwaltung anhand von Aufgabennetzen.** Studienarbeit; TU Berlin, September 1988

Reinhard Keil-Slawik
 TU Berlin, FB Informatik
 Sekr. FR 5-6
 Franklinstraße 28/29
 D-1000 Berlin 10