

Ansätze zur Kooperationsmodellierung im Vergleich

Diplomarbeit am Fachbereich Informatik,
Arbeitsbereich Softwaretechnik der Universität Hamburg
Juni 2000

von Nilgün Özek

Erstbetreuer : Dr. Ralf Klischewski

Zweitbetreuer : Prof. Dr. Horst Oberquelle

Inhaltsverzeichnis

| | |
|---|-----------|
| ABBILDUNGSVERZEICHNIS | 4 |
| TABELLENVERZEICHNIS..... | 5 |
| 1 EINLEITUNG | 6 |
| 1.1 PROBLEMSTELLUNG UND MOTIVATION | 6 |
| 1.2 METHODIK DER ARBEIT | 8 |
| 1.3 GLIEDERUNG DER ARBEIT | 9 |
| 2 KOOPERATIONSMODELLIERUNG..... | 11 |
| 2.1 KOOPERATION..... | 11 |
| 2.2 MODELLIERUNG IM RAHMEN DER SOFTWAREENTWICKLUNG..... | 14 |
| 2.3. KOOPERATION ALS GEGENSTAND DER MODELLIERUNG IM RAHMEN DER ANWENDUNGSENTWICKLUNG | 16 |
| 2.5 SOFTWAREENTWICKLUNG UND ORGANISATIONSENTWICKLUNG..... | 18 |
| 2.6 ANFORDERUNGEN AN DIE KOOPERATIONSMODELLIERUNG | 21 |
| 2.6.1 Sichtweisen der Kooperationsmodellierung..... | 21 |
| 2.6.1.1 Organisationsorientierte Sichtweise | 22 |
| 2.6.1.2 Arbeitsorientierte Sichtweise | 25 |
| 2.6.1.3 Formalisierungsorientierte Sichtweise | 27 |
| 2.6.1.4 Pragmatikorientierte Sichtweise | 28 |
| 3 ANSÄTZE ZUR KOOPERATIONSMODELLIERUNG - EIN VERGLEICHENDER ÜBERBLICK 30 | |
| 3.1 AUSWAHL DER METHODEN /SYSTEMATIK DES VERGLEICHS | 30 |
| 3.2 ANSÄTZE ZUR KOOPERATIONSMODELLIERUNG..... | 32 |
| 3.2.1 RFA-NETZE..... | 32 |
| 3.2.2 SEEME | 37 |
| 3.2.3 ARIS | 44 |
| 3.2.4 ACTION/WORKFLOW..... | 52 |
| 3.3 ERGEBNISSE DES VERGLEICHS..... | 58 |

| | |
|--|-----------|
| 4 VERGLEICH DER ANSÄTZE AM FALLBEISPIEL "PRÜFUNGSLEISTUNG DIPLOMARBEIT" | 60 |
| 4.1. FALLBEISPIEL: "PRÜFUNGSLEISTUNG DIPLOMARBEIT" | 60 |
| 4.2 MODELLIERUNG MIT DEN ANSÄTZEN..... | 63 |
| 4.2.1 Modellierung mit RFA-Netzen | 63 |
| 4.2.2 Modellierung mit SeeMe | 67 |
| 4.2.3 Modellierung mit ARIS | 70 |
| 4.2.4 Modellierung mit Action/Workflow | 72 |
| 4.3 VERGLEICH DER ANSÄTZE..... | 76 |
| 5 ERGEBNISSE/FAZIT | 80 |
| 5.1 BEWERTUNG DER ANSÄTZE ZUR KOOPERATIONSMODELLIERUNG | 80 |
| 5.2 ZUSAMMENFASSUNG UND AUSBLICK..... | 84 |
| LITERATURVERZEICHNIS | 87 |

Abbildungsverzeichnis

| | |
|--|----|
| ABBILDUNG 1: SPANNUNGSFELD ZWISCHEN ORGANISATIONSENTWICKLUNG UND SOFTWAREENTWICKLUNG | 7 |
| ABBILDUNG 2: WESENTLICHE ASPEKTE VON KOOPERATION..... | 13 |
| ABBILDUNG 3: SICHTWEISEN DER KOOPERATIONSMODELLIERUNG..... | 22 |
| ABBILDUNG 4: AUSGEWÄHLTE ELEMENTE DER RFA-NOTATION | 34 |
| ABBILDUNG 5: ELEMENTE DER SEEMe-NOTATION | 39 |
| ABBILDUNG 6: VORGEGEBENE (STANDARD)BEDEUTUNG VON RELATIONEN ZWISCHEN BASELEMENTEN | 40 |
| ABBILDUNG 7: BEISPIEL FÜR VAGES MODELLIEREN | 43 |
| ABBILDUNG 8: SICHTEN DES ARIS-HAUSES..... | 48 |
| ABBILDUNG 9: ACTION/WORKFLOW-LOOP | 55 |
| ABBILDUNG 10: KOOPERATION ZWISCHEN ROLLEN (R-NETZ) | 64 |
| ABBILDUNG 11: KONTROLLNETZ MIT ROLLENZUORDNUNG (RA-NETZ) | 65 |
| ABBILDUNG 12: AKTIONSNETZ ALS ÜBERLAGERUNG VON OBJEKTNETZ UND KONTROLLNETZ..... | 67 |
| ABBILDUNG 13: MODELLIERUNG DES FALLBEISPIELS „PRÜFUNGSLEISTUNG DIPLOMARBEIT“ MIT SEEMe..... | 68 |
| ABBILDUNG 14: MODELLIERUNG DES FALLBEISPIELS "PRÜFUNGSLEISTUNG DIPLOMARBEIT" MIT eEPKS | 71 |
| ABBILDUNG 15: FALLBEISPIEL "PRÜFUNGSLEISTUNG DIPLOMARBEIT" AUFGETEILT IN PHASEN UND ROLLEN | 74 |
| ABBILDUNG 16: MODELLIERUNG DES FALLBEISPIELS "PRÜFUNGSLEISTUNG DIPLOMARBEIT" MIT ACTION/WORKFLOW..... | 75 |
| ABBILDUNG 17: ADONIS-CUSTOMIZING | 79 |

Tabellenverzeichnis

| | |
|---|----|
| TABELLE 1: ELEMENTE DER ERWEITERTEN EREIGNISGESTEUERTEN PROZESSKETTEN | 51 |
| TABELLE 2 : NAMEN, KURZNAMEN DER ANSÄTZE MIT LITERATURHINWEISEN | 58 |
| TABELLE 3: EINSATZKONTEXT/ENTSTEHUNGSGESCHICHTE UND THEORETISCHER FUNDIERUNG DER ANSÄTZE. | 58 |
| TABELLE 4 : GEGENÜBERSTELLUNG DER BASELEMENTE UND IHRE VERKNÜPFUNGEN | 59 |
| TABELLE 5: FALLBEISPIEL "PRÜFUNGSLEISTUNG DIPLOMARBEIT" | 63 |
| TABELLE 6: SPRECHAKTE IM FALLBEISPIEL "PRÜFUNGSLEISTUNG DIPLOMARBEIT" | 73 |

Kapitel 1

Einleitung

1.1 Problemstellung und Motivation

Im Kontext von Arbeit und Organisationen reicht heute nicht mehr aus, nur die individuelle Arbeit zu unterstützen. Die Entwicklung und Einführung von Anwendungssystemen für die kooperative Arbeit stellen neue Herausforderungen an die Softwareentwicklung und SoftwareentwicklerInnen in der Praxis. Der Einsatz von Anwendungssystemen in Organisationen ist gleichbedeutend mit einer Wechselwirkung von Software, menschlicher Tätigkeit und Aufgaben der Organisation. Die Softwareentwicklung für kooperative Arbeitsprozesse dient hierbei einerseits zur Unterstützung dieser Prozesse andererseits jedoch greift sie auch in die sozialen Zusammenhänge, Arbeitsprozesse und Organisationsstrukturen. Es setzt sich in diesem Zusammenhang zunehmend die Erkenntnis durch, daß eine integrierte Sichtweise von Mensch, Organisation und Technik mit einer adäquat verzahnten Vorgehensweise, eine Voraussetzung für erfolgreiche Softwareentwicklungs- und -einführungsprojekte ist.

Die enge Verflechtung von Softwareentwicklung einerseits und die Gestaltung kooperativer Arbeit in Organisationen andererseits führt dazu, daß ein Bedarf nach Ansätzen zur Modellierung entsteht, die sowohl für die Organisationsentwicklung und Arbeitsgestaltung als auch für die Softwareentwicklung eine Hilfestellung bieten. Diesen Zusammenhang bringt folgender Abbildung zum Ausdruck.

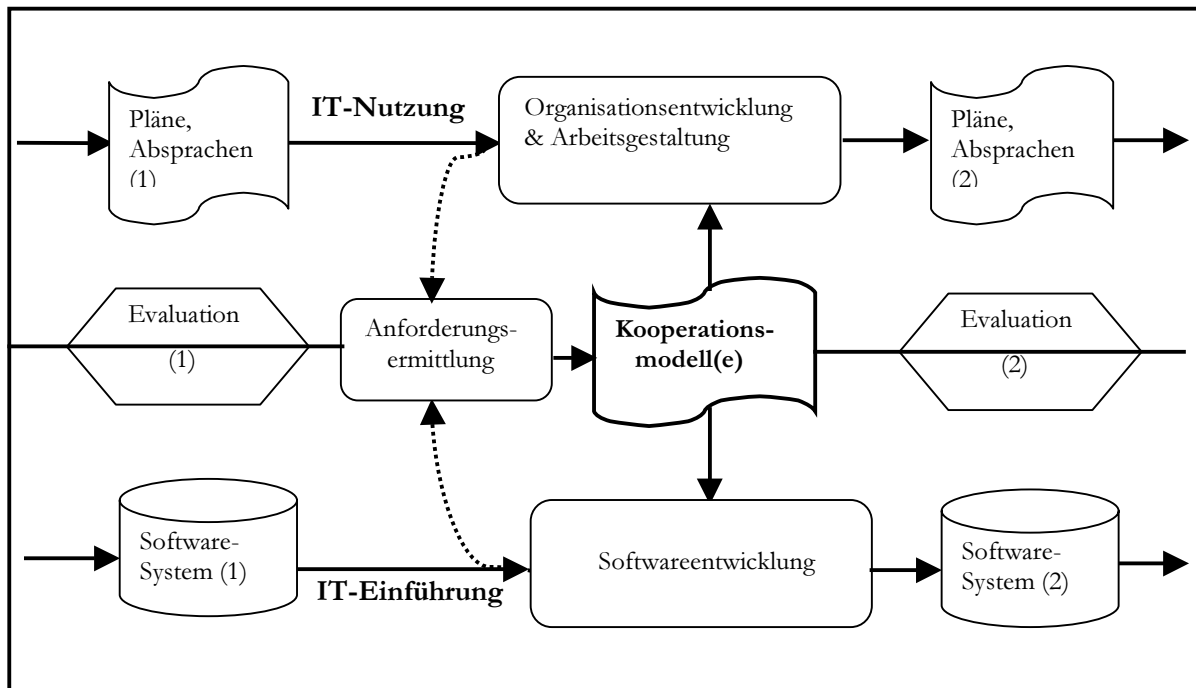


Abbildung 1: Spannungsfeld zwischen Organisationsentwicklung und Softwareentwicklung

Sowohl in der Wirtschaft als auch in der Wissenschaft sind Ansätze entstanden, die sich mit verschiedenen Aspekten dieses Problems beschäftigen. Dabei fließen Gesichtspunkte verschiedener Disziplinen wie Wirtschaftswissenschaften, Arbeitspsychologie, Soziologie und Informatik ein. Es ist daher auch nicht erstaunlich, daß die vorhandenen Ansätze sich durch ihre Sichtweisen auf diese Problematik unterscheiden. Diese verschiedenen Sichtweisen stellen entweder die Softwareentwicklung, die Organisationsentwicklung oder Arbeitsgestaltung in den Vordergrund bei der Modellierung. Da die ausgewählten Methoden auch die Sichtweise auf den Gegenstandsbereich bestimmen, müssen sich die SoftwareentwicklerInnen auch mit den Sichtweisen der ausgewählten Methoden auseinandersetzen, damit sie die beabsichtigte Art der Modellierung herausarbeiten, und so an das Problem bewußter herangehen können.

Es wird in dieser Arbeit unter Kooperationsmodellierung, Ansätze im Rahmen der Anwendungsentwicklung, die im Modell wesentliche Aspekte von Kooperation - sowohl visuell als auch konzeptionell - explizit machen, verstanden. Es werden vier Ansätze zur Kooperationsmodellierung ausgewählt und verglichen. Diese sind im einzelnen:

1. Rollen-/Funktions-/Aktionsnetze (RFA-Netze) von Horst Oberquelle
2. Soziorientierte semistrukturierte Modellierungsmethode (SeeMe) von Thomas Hermann et al.
3. Architektur integrierter Informationssysteme (ARIS) von August Wilhelm Scheer
4. Action/Workflow von Terry Winograd/Fernando Flores

Ausgehend aus der Definition von Kooperationsmodellierung und aus der Problemstellung ergeben sich für diese Diplomarbeit folgende Fragestellungen:

1. Welche Aspekte von Kooperation werden durch diese Ansätze im Modell - sowohl konzeptionell als auch visuell - explizit gemacht?
2. Wie unterstützen diese Ansätze die Integration zwischen Softwareentwicklung einerseits Organisationsentwicklung und Arbeitsgestaltung andererseits?

1.2 Methodik der Arbeit

Diese Diplomarbeit hat das Ziel, durch die Beantwortung dieser Fragen, eine Orientierungshilfe für die SoftwareentwicklerInnen bei der Auswahl der Methoden zur Kooperationsmodellierung bieten. Zu diesem Zweck werden verschiedene Sichtweisen der Kooperationsmodellierung herausgearbeitet, die ausgewählten Ansätze, sowohl analytisch als auch praktisch verglichen und abschließend bewertet.

Bei der Auswahl der Ansätze ist insbesondere darauf geachtet, daß die Ansätze explizite Konzepte von Kooperation aufweisen, graphische Modellierung ermöglichen und einen Metamodell zur Verfügung stellen. Für die SoftwareentwicklerInnen spielt, die in den letzten Jahren aktuell gewordene Modellierungssprache UML (Unified Modeling Language) eine große Rolle. Trotz dieser Bedeutung der UML für die SoftwareentwicklerInnen, ist sie nicht für den Vergleich ausgewählt. Ein wesentlicher Grund hierfür war, daß die UML konzeptionell sich nicht explizit mit Kooperation beschäftigt. Sie leistet mit ihren benutzerorientierten Sichten wie Use-Cases und User-Aktivitätsdiagrammen eine große Hilfe bei der Analyse und Design von Softwaresystemen. Sie bietet jedoch keine angemessene Unterstützung bei der Organisationsentwicklung und Arbeitsgestaltung. Die fehlenden expliziten Konzepte von Kooperation war auch der Grund, warum der Ansatz ADONIS - obwohl sie für die praktische Entwicklungsarbeit eine große Hilfe leistet - nicht in die Liste der Ansätze zur Kooperationsmodellierung aufgenommen ist.

Der analytische Vergleich der ausgewählten Ansätze wird nach einem vordefinierten Raster durchgeführt. Es werden dabei fünf Kategorien der Analyse unterschieden:

1. Entstehungsgeschichte
2. Theoretische Fundierung
3. Basiselemente und ihre Verknüpfung
4. Einsatzkontext

Zum Zwecke des praktischen Vergleichs wird das Fallbeispiel "Prüfungsleistung Diplomarbeit" ausgewählt und mit jedem Ansatz getrennt modelliert. Ein wesentlicher Grund für die Auswahl dieses Fallbeispiels war, daß in ihm eine Vielzahl von Personen (Studierende, Betreuer bzw. Gutachter, Prüfungsamt bzw. entsprechende SachbearbeiterInnen in den Fachbereichen) beteiligt sind, und die Kooperation zwischen den Beteiligten auf unterschiedliche Art und Weise - sowohl formal als auch informell - erfolgt. Während beispielsweise, die Interaktion zwischen Studierenden bzw. Betreuer einerseits und der Prüfungsverwaltung andererseits formal geregelt abläuft, ist die Kooperation innerhalb eines Betreuungsverhältnisses von eher informellem und individuellem Charakter.

Eine systematischer, detaillierter und nachvollziehbarer Methodenbewertung erfordert die Definition von methodenunabhängigen Bewertungskriterien. Aus diesem Grund werden zunächst - ausgehend aus der Problemstellung und aus der Beobachtung in Softwareprojekten - vier Sichtweisen herausgearbeitet, die für die Kooperationsmodellierung eine Notwendigkeit darstellen. Diese sind im einzelnen:

1. Organisationsorientierte Sichtweise
2. Arbeitsorientierte Sichtweise
3. Formalisierungorientierte Sichtweise
4. Pragmatikorientierte Sichtweise

Aus diesen Sichtweisen heraus werden die Anforderungen an die Kooperationsmodellierung definiert. Die für den Vergleich ausgewählten Ansätze, werden dann anhand dieser Anforderungen aus verschiedenen Sichtweisen der Kooperationsmodellierung bewertet.

1.3 Gliederung der Arbeit

Der erste Kapitel dieser Arbeit besteht aus dieser Einleitung, das die Problemstellung, Fragestellung und Methodik der Arbeit beinhaltet.

Im zweiten Kapitel wird zunächst der Begriff "Kooperation" näher erläutert. Danach wird auf die Modellierung im Rahmen der Softwareentwicklung eingegangen. Anschließend wird die Kooperation als Gegenstand der Modellierung im Rahmen der Anwendungsentwicklung betrachtet. Nachfolgend wird die Softwareentwicklung und Organisationsentwicklung thematisiert. Zum Schluß werden die verschiedenen Sichtweisen der Kooperationsmodellierung dargestellt und die Anforderungen an die Kooperationsmodellierung aus diesen Sichtweisen herausgearbeitet.

Im Kapitel drei werden die ausgewählten Ansätze zur Kooperationsmodellierung vergleichend dargestellt. Zunächst werden die Kriterien für die Auswahl vorgestellt und die Auswahl der Ansätze begründet. Danach werden die Ansätze nach einem vordefinierten Raster analytisch verglichen. Schließlich werden die Ergebnisse des analytischen Vergleich tabellarisch zusammengefaßt.

Der praktische Vergleich der Ansätze bildet den Gegenstand des vierten Kapitels. Hierbei wird mit jedem Ansatz das Fallbeispiel "Prüfungsleistung Diplomarbeit" graphisch modelliert. Anschließend werden die Ansätze im Hinblick auf die Erstellung der komplexer Modelle verglichen.

Im fünften Kapitel werden zunächst die Ansätze anhand von Anforderungen aus den verschiedenen Sichtweisen der Kooperationsmodellierung bewertet. Anschließend wird die Arbeit zusammengefaßt und weitere Forschungsarbeiten, die sich aus den Ergebnissen dieser Arbeit ergeben, angesprochen.

Kapitel 2

Kooperationsmodellierung

2.1 Kooperation

In der Informatik wird der Begriff "Kooperation" insbesondere im Zusammenhang mit der Computer-Supported-Cooperative-Work (CSCW) benutzt, der sich mit der Kooperationsunterstützung beschäftigt. Obwohl dieses Gebiet sich seit vielen Jahren mit diesem Thema beschäftigt, findet sich keine eindeutige Definition in der CSCW-Literatur.

Der Begriff "Kooperation" wird in [Pie91] wie folgt definiert:

"Kooperation bezeichnet das Tätigsein von zwei oder mehr Individuen, das bewußt planvoll aufeinander abgestimmt die Zielerreichung eines jeden beteiligten Individuums in gleichem Maße gewährleistet."
([Pie91] S.81)

Bannon et al. weisen darauf hin, daß die kooperative Arbeit nicht schlechter oder besser als die individuelle Arbeit betrachtet werden sollte, sondern als die Notwendigkeit für die Erledigung einer Aufgabe, die individuell nicht zu erledigen ist [BS89]. Das Motiv kooperativer Arbeit wird von Bannon et al. wie folgt benannt:

„[...] People engage in cooperative work when they are mutually dependent in their work and therefore are required to cooperate in order to get the work done“ ([BS89], S. 7)

Eine allgemeine Definition von kooperativer Arbeit wird von Oberquelle geliefert.

"Unter kooperativer Arbeit sollen Arbeitssituationen verstanden werden, in denen mehrere Personen als Gruppe zusammenarbeiten zwecks Erreichung eines Ergebnisses, welches nur gemeinsam, aber nicht einzeln erreicht werden kann." ([Obe91], S. 4)

Um kooperativer Arbeit von nicht-kooperativer Arbeit zu unterscheiden, sind nach Oberquelle folgende Eigenschaften bestimmend:

- *"Mindestens partielle Übereinstimmung der Ziele der beteiligten Personen,*
- *Gemeinsame Nutzung knapper Ressourcen durch Austausch oder gleichzeitige Nutzung,*
- *Koordination der Einzelhandlungen gemäß vereinbarten Konventionen,*
- *Verständigung über Ziele und Konventionen der Zusammenarbeit zwecks flexibler Anpassung"*
([Obe91], S. 4)

Die Kooperation kann vielfältige Formen annehmen. Eine Unterscheidung wird von Piepenburg unternommen[Pie91]:

Bilaterale vs. multiple Kooperation: Hierbei bezieht sich diese Unterscheidung auf die Anzahl der beteiligten Kooperationspartner. Bilateral bedeutet dabei, daß die Kooperation unter Beteiligung von zwei Partnern stattfindet. Wenn an der Kooperation mehr als zwei Personen beteiligt sind, wird dann von multipler Kooperation gesprochen.

Konjunktive vs. disjunktive Kooperation: Diese Dimension bezieht sich auf die Art und Weise, in der die Kooperationspartner das Arbeitsergebnis erzielen. Wirken die beteiligten Akteure mit eigenen, eindeutig definierten Handlungen auf den Handlungsablauf bis hin zur Zielerreichung, so wird von konjunktiver Kooperation gesprochen. Verfolgt hingegen jeder Akteur vergleichbare individuelle Handlungspläne und ist es lediglich erforderlich, daß ein Akteur bis zum antizipierenden Ziel vordringt, wird dies als disjunktive Kooperation bezeichnet.

Unmittelbare vs. mittelbare Kooperation: Diese Dimension bezieht sich auf die räumliche und/oder zeitliche Distanz der Kooperationspartner. Unmittelbare Kooperation ist immer dann gewährleistet, wenn die Akteure die Möglichkeit einer direkten, nicht ausschließlich technisch vermittelnden Kommunikation haben. Mittelbare Kooperation bezieht sich auf alle übrigen Formen, d.h.

zeitversetzt, aber am gleichen Ort (z.B. Schichtarbeit), zeitgleich, an verschiedenen Orten; sowie zeitversetzt, an verschiedenen Orten.

Kooperative Arbeitsprozesse sind meistens in einem organisatorischen Umfeld eingebettet, z.B. in einem Unternehmen, einem Krankenhaus, einer Universität oder einer Behörde. Unterschiedliche kooperativer Prozesse laufen im allgemeinen nicht isoliert voneinander ab, sondern sind durch Datenaustausch, Kommunikation, gemeinsamer Aufgabenträger oder gemeinsam genutzte Ressourcen miteinander verbunden [Obe96].

In diesem Zusammenhang wird in der Literatur auf die enge Verflechtung zwischen Menschen(gruppen), Arbeitsaufgaben, Organisationsstrukturen, Technologie und die Umgebung hingewiesen ([Obe91], [Obe94], [PM98], [HSW98]).

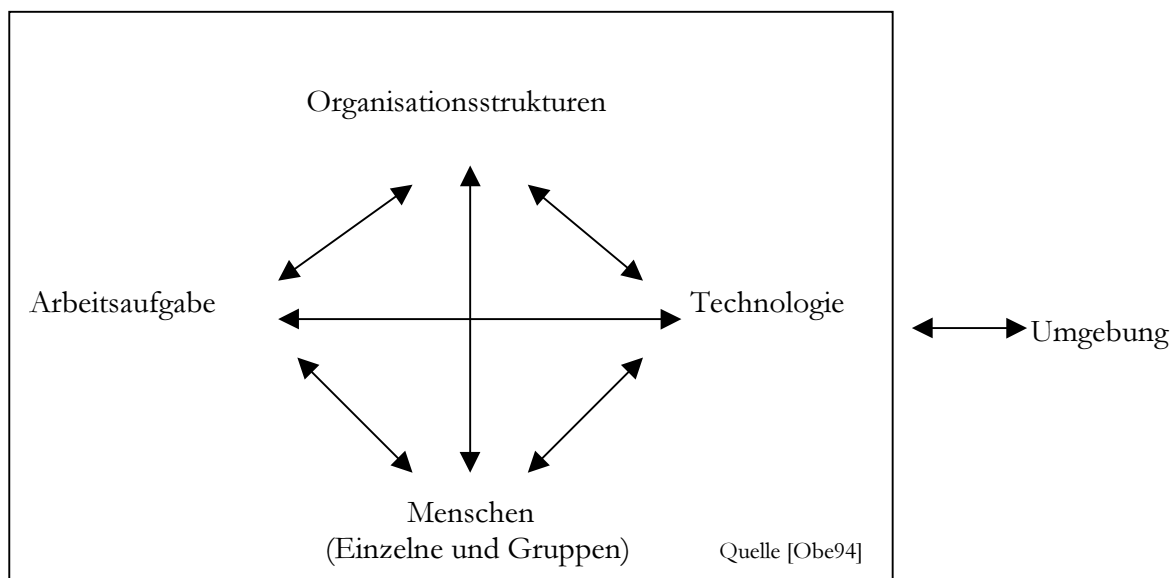


Abbildung 2: Wesentliche Aspekte von Kooperation

Diese Abbildung soll darauf hinweisen, daß bei der Betrachtung von Kooperation fünf Aspekte (Menschen mit ihren Eigenschaften, die Arbeitsaufgaben, die Organisationsstrukturen, die verfügbare Technologie und Umgebung) sich gegenseitig bedingen und beeinflussen. Es wird in diesem Zusammenhang darauf hingewiesen, daß diese Aspekte bei der Analyse von Kooperationen ganzheitlich wahrgenommen und in ihrem Zusammenspiel verstanden werden müssen ([Obe91], [Obe94]).

2.2 Modellierung im Rahmen der Softwareentwicklung

Im Rahmen der Softwareentwicklung kommt der Modellierung und den Modellen eine zentrale Stellung zu. Die Bedeutung der Modelle für die Softwareentwicklung bringt folgendes Zitat zum Ausdruck:

„Modelle sind zentrale Hilfsmittel für Systemanalyse, Algorithmenentwicklung und Systemgestaltung. Modelle bestimmen die Sichtweisen des (zukünftigen) Anwendungsbereichs für die gesamte Systementwicklung und –nutzung –einerseits durch das, was im Modell (nicht) enthalten ist, andererseits auch durch die Eigenschaften, die Modellen in ihrer Verwendung beigemessen wird.“ ([Kli96], S. 101)

Aus Sicht der Softwaretechnik ist die Softwareentwicklung - aufgrund der Erstellung unterschiedliche ausführbare und nicht-ausführbare Modelle - ein Modellierungsprozeß [Zül98]. Es geht dabei, um verschiedene Arten und Ebenen der Modellierung und sogar um verschiedene Modellbegriffe [FKW98]. Modelle im Rahmen der Softwareentwicklung können zur algorithmischen Realisierung, zur formalen Spezifikation (Petri-Netze) und zum Verständnis des Anwendungsbereichs dienen. Floyd und Klischewski unterscheiden in diesem Zusammenhang folgende Ebenen der Modellierung im Rahmen der Softwareentwicklung:

- das Anwendungsmodell (Gegenstandsbereich)
- das formale Modell (Spezifikation)
- das Berechnungsmodell (Programm).

Beim Anwendungsmodell geht es um idealisierte oder konkrete Darstellung des Gegenstandes. Beim formalen Modell wird das Anwendungsmodell zu Satz- und Formelmengen zugeordnet. Das Berechnungsmodell wird durch eine Menge von zusammenwirkenden berechenbaren Funktionen definiert. Das formale Modell soll sich einerseits in einem Berechnungsmodell umsetzen lassen, und gleichzeitig in einem engen Zusammenhang mit der Anwendungswelt stehen [WKK99]. Mit dieser Art von Modellierung sind viele Fragestellungen verbunden, die im Rahmen der Softwareentwicklung zu beachten sind [FK98]: Wie verhält sich das Anwendungsmodell zum Gegenstandsbereich? Wie können die einzelnen Elemente im Rahmen der Softwareentwicklung ineinander überführt werden? Wie verhält sich das Berechnungsmodell zur technischen bzw. sozialen Wirklichkeit des Einsatzkontextes?

Floyd und Klischewski charakterisieren die Aktivitäten, die bei der Modellierung im Rahmen der Softwareentwicklung maßgeblich sind, wie folgt:

- *"Informatisieren*, d.h. materielle oder ideelle Gegenstände als Informationsträger zu betrachten, aus deren Aspekten die für die Operation relevanten Daten ermittelt werden können;
- *Diskretisieren*, d.h. ausgewählte Aspekte der Gegenstände durch diskrete Wertebereiche zu charakterisieren, Operationen auf diesen Wertebereichen zu definieren sowie Zustände, die durch Ausführung von Operationen transformiert werden;
- *Systemisieren*, d.h. ein Informatiksystem aus zu modellierenden Elementen, Beziehungen und Operationen sowie dessen Wechselwirkung mit dem (häufig ebenfalls als System betrachteten) Kontext zu konzipieren." ([FK98], S. 22)

Diese Vorgehensweise wird operationale (Re)Konstruktion genannt. Es wird in diesem Zusammenhang argumentiert, daß bei der Modellierung im Rahmen der Softwareentwicklung, der jeweils interessierende Gegenstandsbereich als operationaler Form betrachtet wird:

"Dabei werden menschliche Handlungen, natürliche oder technische Vorgänge als Operationen charakterisiert und logisch, kausal oder zeitlich verknüpft. Operationale Form beginnt mit der Beschreibung von zur Routine gewordenen Handlungen und führt allmählich zur Vergegenständlichung in symbolischen und technischen Artefakten. Durch informatischer Modellierung wird ein interessierender (natürlicher, technischer, sozialer oder ideeller) Gegenstandsbereich operational (re) konstruiert." ([FK99], S. 4)

Dabei weisen Floyd und Klischewski auf die Schwierigkeiten hin, die sich durch diese Art von Modellierung ergeben. Sie betonen dabei, daß es bei der Modellierung im Rahmen der Softwareentwicklung, nicht nur darum geht, wie die Realität aufgenommen wird, sondern auch um ihre tatsächliche Wirksamkeit in dem eingesetzten Kontext:

"Obwohl Informatik-Modelle selbst formal sind und technisch realisiert werden, werden sie nur bedeutsam durch soziale Prozesse der Entwicklung und Aneignung, wobei die subjektiven Perspektiven der Beteiligten entscheidenden Einfluß haben. Entwicklung und Nutzung von Informatik-Modellen sollten daher ihre sozialen Bedingtheit transparent machen und ihre sinnvolle Interpretation im Einsatzkontext gewährleisten" ([FK98], S.21)

Die Modellierung im Rahmen der Softwareentwicklung wird daher einerseits als "Fenster zur Wirklichkeit" im Sinne von Perspektiven der Wahrnehmung, und andererseits im Hinblick auf ihre intendierte Wirksamkeit beim Einsatz als "Handgriff zur Wirklichkeit" charakterisiert. [FK98], [WKK99].

2.3. Kooperation als Gegenstand der Modellierung im Rahmen der Anwendungsentwicklung

Wenn Kooperation als Gegenstand der Modellierung im Rahmen der Softwareentwicklung betrachtet wird, stellt sich die für das Anwendungsmodell erforderliche Identifikation von Attributen, Operationen, Zustände und zugehörigen Transformationen problematisch. Kooperationsbeziehungen sind soziale Beziehungen und sind von daher nicht als solches modellierbar. Die Betrachtung der Kooperation als Modellgegenstand bedeutet jedoch, daß nur diejenigen Aspekte von Kooperation, die in den Modellen abgebildet werden können - formalisierbaren, wiederkehrenden bzw. verallgemeinerbaren Anteile -, bei der Modellierung betrachtet werden. Was passiert jedoch mit den kooperativen Anteilen der Arbeit, die zur Formalisierung nicht zugänglich sind? Wie soll damit im Rahmen der Softwareentwicklung umgegangen werden? Softwareentwicklung befindet sich hiermit in einem Spannungsfeld zwischen dem formalen, technischen Modell (programmierter Struktur) und nichtformalen, sozialen Einsatzkontext[Fuc92], [Val97]. Für die Anwendungsentwicklung ist es jedoch unerlässlich, daß die Integration zwischen dem formalen einerseits und nichtformalen andererseits sowohl theoretisch als auch praktisch bewältigt wird, da beim Einsatz von Software für die Kooperation nicht nur eine Verlagerung von Tätigkeiten zwischen Mensch und Maschine, sondern auch zwischen Menschen(gruppen) untereinander stattfindet.

Es muß der Kontext, der für das Verständnis und für die Anwendung des Modells maßgeblich ist, bei der Modellierung mitbetrachtet werden. Dies bedeutet gleichzeitig, daß die Modellierung als einen sozialen Prozeß verstanden werden muß. Floyd und Klischewski stellen von daher, die über Informatisierung, Diskretisierung und Systemisierung hinausgehende Anforderungen an die Modellierung im Rahmen der Anwendungsentwicklung[FK98, S.25]:

- als Ausgangspunkt der Modellierung soll nicht die "objektive Welt", sondern die Perspektivität der beteiligten Akteure anerkannt werden.
- Modellbildung soll als sozialen Prozeß verstanden werden.

- um die spätere Aneignung von Modellen zu erleichtern, soll der Verlauf der sozial bedingten Gestaltungsentscheidungen im Rahmen der Modellierung transparent bleiben.
- Der Einsatz von Informatik-Modellen soll so gestaltet werden, daß in der Situation Raum für verantwortliches Handeln und Veränderung verbleibt.

Diese Anforderungen stellen an die SoftwareentwicklerInnen neue Herausforderungen. Bei der Betrachtung von Kooperation als Gegenstand der Modellierung müssen die SoftwareentwicklerInnen neben den technischen Problemen, sich auch mit den nicht-technischen Problemen - wie Organisationsentwicklung, Arbeitsgestaltung - im Rahmen der Modellierung beschäftigen:

"(...) Here, software developers provide tools for the work for users and media for the communication between users. Moreover, software is used for coordinating work and thus embodies concepts for organizational control. Thus, on one hand, design relies on cooperation and negotiation amongst different parties: developers, users, management and so on, and the technical concerns for constructing high quality products are inherently tied up with communication and work - social processes which define the very nature of the problems to be dealt it. On the other hand, the systems in use shape the conditions for organizational development." ([Flo99], S.5)

Es stellt sich in diesem Zusammenhang die Frage, welche Ansätze hierbei eine Hilfestellung bieten. Das Problem der Integration der Softwareentwicklung einerseits und Organisationsentwicklung und Arbeitsgestaltung andererseits erfordert die Untersuchung verschiedene Arten von Modellierung.

Für die Softwareentwicklung spielen die strukturierte- und insbesondere objektorientierte Methoden eine große Rolle. Sie weisen jedoch bei der Analyse und Modellierung kooperativer Arbeitsprozesse Schwachpunkte:

"Standard system development methods prescribe representational techniques aimed at rendering working practices into forms narrowly to design concerns." ([Suc95], S. 33)

Die objektorientierte Methoden, insbesondere, die in den letzten Jahren aktuell gewordene Modellierungssprache UML (Unified Modeling Language) sind in der Praxis für die SoftwareentwicklerInnen unentbehrlich. Sie leistet mit ihrem benutzerorientierten Sichten, solche wie Use-Cases und User-Aktivitätsdiagrammen eine große Hilfe bei der Modellierung von Softwaresy-

stemem. Was jedoch im Hinblick auf die Kooperationsmodellierung zu bemängeln ist, daß die Modellierungssprache UML kein Konzept von Kooperation besitzt, und keine angemessene Unterstützung bei der Arbeits- und Organisationsgestaltung anbietet.

"They are an effective way to document work processes but do not support problem identification, improvement analysis (...)." ([BEH99], S.39)

Dies ist auch nicht erstaunlich, da bei der Entwicklung von UML die Softwareentwicklung im Vordergrund der Interesse stand. Auch die Worte von Jacobson bringen dies zum Ausdruck:

"Use-Cases sind dazu da, beim Auffinden der funktionalen Anforderungen zu helfen. (...) Unter Geschäftsprozeßmodellierung verstehe ich das Auffinden von Anforderungen an das Softwaresystem (...)" ([BJ97], S. 22)

Die Aktivitätsdiagramme sind grundsätzlich geeignet, um Geschäftsprozesse zu beschreiben und zu modellieren. Als Schwachpunkte können jedoch in diesem Zusammenhang die Schwierigkeiten bei der Abbildung von Organisationsstrukturen und der Modellierung von Rollenkonzepten und Berechtigungen angegeben werden ([Oes98], [Wal98]).

Die ethnographische Methoden wie die Videoaufnahmen werden in letzter Zeit im Rahmen der Softwareentwicklung eingesetzt. Diese Methoden sind speziell dazu gedacht, Gruppen und das sie umgebende (Arbeits-)Umfeld zu untersuchen. Mit Videoeinsatz sollen die Personen und die Arbeit beobachtet werden, um so einen Eindruck von der Arbeit zu bekommen.

Der Einsatz ethnographischer Methoden bringt jedoch bei der Modellierung die Frage mit sich, wie die Ergebnisse der ethnographischer Studien, in die Softwareentwicklung zu übertragen sind. Da der Übergang von der Analyse zur Erstellung von Software bei dieser Art von Modellierung offenbleibt, können die SoftwareentwicklerInnen mit dieser rein beschreibenden Art von Modellierung wenig anfangen.

2.5 Softwareentwicklung und Organisationsentwicklung

Es wird sowohl in der Literatur als auch in der Praxis öfters zum Ausdruck gebracht, daß die Entwicklung und Einführung von Anwendungssystemen für die kooperativen Arbeitsprozesse im Kontext von Organisationsentwicklung und Arbeitsgestaltung zu betrachten sind [Rol98],

[GM97], [PM98], [FKR97], [KWJ99], [HSW98]. Dies stellt jedoch eine Herausforderung für die SoftwareentwicklerInnen in der Praxis:

"Dealing with change is one of the most fundamental challenges facing IS professionals today. (...) In dealing with change, we need to address not only technological issues, such as "How do I build a system that does X?" but also human and social ones, such as "How does a group of people accomplish task Y?" as well as organizational ones, such as "How can we ensure policy Z?" ([DDJ98], S. 64 und S.70)

Es wird auf der anderen Seite jedoch bemängelt, daß die notwendige Integration zwischen Softwareeinführung und Organisationsentwicklung in den Softwareprojekten nicht die angemessene Beachtung findet:

"Die allermeisten Projekte zur Softwareeinführung in Unternehmen zeigen, daß die durch die Technologieeinführung zwangsläufig zu behandelnden Aspekte der Arbeitsgestaltung und Organisationsentwicklung nicht angemessen im Projekt bzw. Projektumfeld thematisiert werden (und daher oft zum Scheitern verurteilt sind)." ([KKL98], S. 81)

In diesem Zusammenhang wird in der Literatur darauf hingewiesen, daß die notwendiger (Weiter)Entwicklung schon bei der Modellierung in Betracht bezogen werden muß ([GMD94]). Diskursive Einbeziehung sowohl individueller als auch kollektiver Anwenderperspektiven wird hierbei unerlässlich.

Es ist in diesem Zusammenhang auch ein Sichtwechsel auf die Modellierung in der Informatik zu beobachten:

"Nicht mehr die Totalität sozialer Organisationen ist Gegenstand der informatischen Modellierung, sondern das Zusammenwirken eingesetzter Informationstechnik mit der sozialen Praxis ist der Gegenstand einer sozial bewerkstelligten, informatischer Modellierung" ([KKM99], S. 304)

Auf die allgemeinen Schwierigkeiten der Anwendungsentwicklung in den früheren Phasen der Softwareentwicklung, in denen es um die Abgrenzung, Analyse und Modellierung jeweiligen projektspezifischen Anwendungsbereich geht, wird öfters in der Literatur hingewiesen:

"Eine besondere Schwierigkeit (und Herausforderung) der Anwendungsentwicklung besteht darin, daß an der Arbeit in den früheren Phasen verschiedene Gruppen von Menschen beteiligt sind und miteinander

kooperieren müssen, die von ihrer Ausbildung, ihrer fachlichen Orientierung und ihrem beruflichen Umfeld teilweise sehr unterschiedlich geprägt sind." ([BHK94a], S.44)

Es ergibt sich insbesondere dadurch, daß diese Gruppen (Entwicklern, Projektleitern, künftige Benutzer und andere Fachleute des Anwendungsbereiches) den Anwendungsbereich aus jeweils verschiedenen Blickwinkeln betrachten. Es ist jedoch notwendig, daß trotz dieser unterschiedlichen Sichtweisen ein einheitliches Verständnis des Anwendungsbereichs geschaffen wird, um eine Kommunikations- und Diskussionsgrundlage gewährleisten zu können.

Im Rahmen der Anwendungsentwicklung kommt daher der Anforderungsermittlung und -analyse eine zentrale Stellung zu. Es bedarf hierbei auch ein anderes Verständnis von der Anforderungsermittlung- und -analyse als in herkömmlichen Sinne. Anforderungsermittlung sollte als

"perspektivischer und dynamischer menschlicher Erkenntnisprozeß (...), in dessen Verlauf sich Anforderungen auf 'emergente' Weise, d.h. dynamisch im Laufe der sozialen Interaktion zwischen den Beteiligten, z.B. den Systemanalytikern und der Organisation ergeben und deshalb auch immer wieder erneuter Überprüfung bedürfen." ([And97], S. 79)

Zur Darstellung und Überprüfung der ermittelnden Anforderungen brauchen die SoftwareentwicklerInnen Methoden, die den Diskurs mit den Akteuren (sowohl individuellen als auch kollektiven) in der Anwendungssituation unterstützen.

„ (...) die dem Verstehen und Handeln aller beteiligten Akteure und nicht nur Systementwickler vor Ort förderlich sind. (...) Neben formalen werden kommunikative Methoden notwendig, die Diskurs und Konsensherstellung bei unterschiedlichen Auffassungen unterstützen.“ ([Rol98], S. 25)

Die erstellten Modelle sollen dabei nicht die Anwendungssituation vollständig spezifizieren, sondern im Entwicklungs- und Einführungsprozeß einen gemeinsamen Lern- und Modellierungsprozeß mit den zukünftigen Anwendern vorantreiben [JKT99].

Nachdem die Modellierung im Rahmen der Softwareentwicklung und Kooperation als Gegenstand der Modellierung im Rahmen der Anwendungsentwicklung betrachtet sowie die enge Verflechtung von Softwareentwicklung und Organisationsentwicklung angesprochen sind, stellt sich

nun die Frage, welche Anforderungen an die Kooperationsmodellierung gestellt werden. Diese Herauszufinden ist der Gegenstand des nächsten Unterkapitels.

2.6 Anforderungen an die Kooperationsmodellierung

In diesem Teil der Arbeit werden die Anforderungen an die Kooperationsmodellierung definiert. Zu diesem Zweck werden - ausgehend aus der Problemstellung und aus den Beobachtungen in den Projekten [FKR97], [KWJ99], [HSW98] - vier Sichtweisen der Kooperationsmodellierung herausgearbeitet.

Bevor die Sichtweisen und ihre Anforderungen an die Kooperationsmodellierung näher beschrieben werden, sollen zunächst die Merkmale der Kooperationsmodellierung beschrieben, und der Begriff "Kooperationsmodellierung" präzisiert werden.

Kooperationsmodellierung zeichnet sich mindestens durch folgende Merkmale aus [KW99]:

- hauptsächlich graphische Beschreibung
- eine spezielle Arbeitssituation/Ort
- Akteure (Individuen, Gruppen, Abteilungen)
- eine oder mehrere Aufgaben, die nur durch Kooperation erledigt werden können.

Diese können durch die Wechselbeziehung in den Akteurshandlungen und deren Ergebnisse beschrieben werden.

In dieser Arbeit wird unter Kooperationsmodellierung, *die Ansätze im Rahmen der Anwendungsentwicklung, die im Modell wesentliche Aspekte von Kooperation - sowohl konzeptionell als auch visuell - explizit machen*, verstanden.

2.6.1 Sichtweisen der Kooperationsmodellierung

"Ein Problem kann und soll nicht isoliert aus einer Perspektive betrachtet werden" ([LHM95], S. 333). Dies gilt auch für das Problem der Integration zwischen Softwareentwicklung einerseits und Organisationsentwicklung und Arbeitsgestaltung andererseits. Es müssen sowohl die Aspekte von Organisationsentwicklung und Arbeitsgestaltung als auch die Aspekte von Formalisierung und der praktische Entwicklungsarbeit bei der Modellierung integriert betrachtet werden. Es werden aus dieser Erkenntnis heraus, im folgenden zwischen vier Sichtweisen der Kooperationsmodellierung (organisationsorientierte-, arbeitsorientierte-, formalisierungsorientierte- und pragmatikorientierte

Sichtweise) unterschieden. Es sei an dieser Stelle ausdrücklich darauf hingewiesen, daß diese Klassifizierung der Sichtweisen keinesfalls als Bewertung verstanden werden sollte. Sie soll vielmehr verdeutlichen, daß diese Sichtweisen alle ihre Berechtigung haben und notwendig für die Kooperationsmodellierung sind. Folgende Abbildung faßt diese vier Sichtweisen der Kooperationsmodellierung mit ihren Untersuchungsobjekten zusammen:

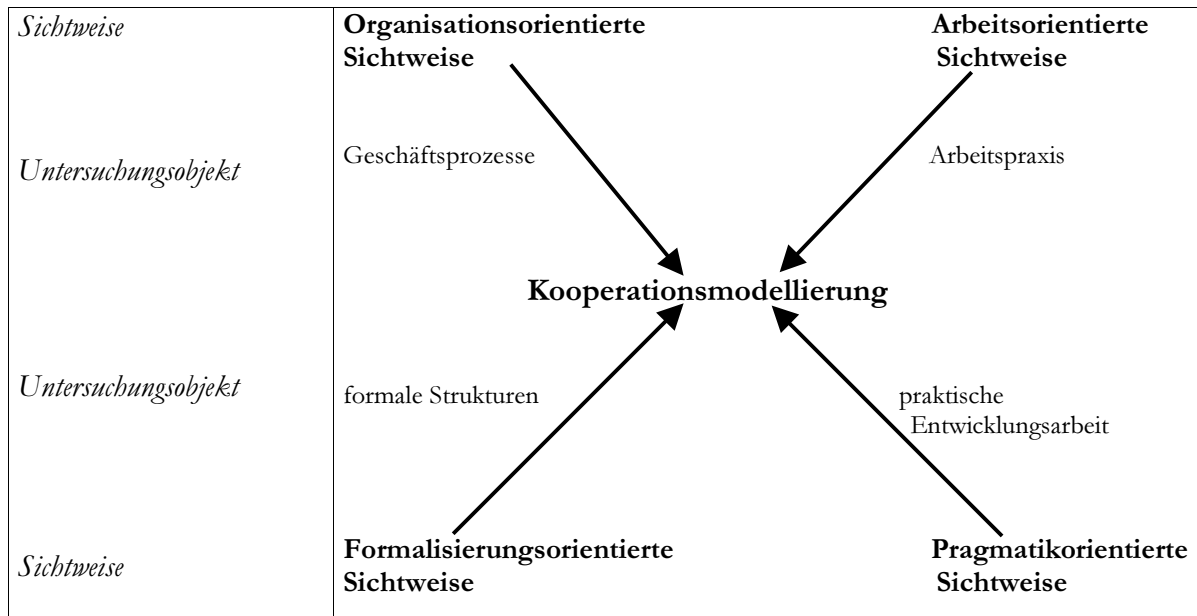


Abbildung 3: Sichtweisen der Kooperationsmodellierung

Im folgenden werden diese Sichtweisen näher aufgeführt und anschließend werden die Anforderungen an die Kooperationsmodellierung aus diesen verschiedenen Sichtweisen definiert, um die ausgewählten Ansätze bewerten zu können.

2.6.1.1 Organisationsorientierte Sichtweise

Diese Sichtweise der Kooperationsmodellierung steht eng mit den Ansätzen zur Organisationsentwicklung (OE) und Organisationslernen (OL) in Verbindung. Bei der Modellierung wird die Organisationsentwicklung und -gestaltung in den Vordergrund der Interesse gestellt. In diesem Zusammenhang stellt, die von Hammer und Champy begründete Prozeßorientierung "Business Process Reengineering" (BPR) eine der Sichtweisen auf die Organisationen, bei der zunächst auf die Unternehmensprozesse (Geschäftsprozesse) das Augenmerk gerichtet wird. Hammer und Champy definieren "Business Process Reengineering" wie folgt:

"Business Reengineering ist fundamentales Überdenken und radikales Redesign von Unternehmen oder wesentlichen Unternehmensprozessen. Das Resultat sind Verbesserungen um Größenordnungen in entscheidenden, heute wichtigen und meßbaren Leistungsgrößen in den Bereichen Kosten, Qualität, Service und Zeit." ([HC94], S.29)

Unter einem Geschäftsprozeß wird dabei, ein Bündel von Aktivitäten, für das eine oder andere unterschiedliche Inputs benötigt werden, und das für den Kunden ein Ergebnis von Wert erzeugt, verstanden. Beispielsweise stellt, die Abwicklung eines Kundenauftrages vom Auftragseingang bis zur Auslieferung, einen Geschäftsprozeß dar. Diese und andere Prozesse sollen zur Aufrechterhaltung von Organisationen dienen und werden daher auch als Gegenstand der Analyse und Modellierung betrachtet.

"Die Modellierung von Geschäftsprozessen ist eine Grundvoraussetzung für die systematische Gestaltung und Verbesserung von Organisationen und die Einführung bzw. Entwicklung prozeßorientierter Systeme." ([NSZ95], S.428)

Bei der Gestaltung der Geschäftsprozesse wird eine abteilungsübergreifende Perspektive eingenommen:

"Geschwindigkeit, Qualität, Abwicklungskosten oder Flexibilität, die immer mehr über die Wettbewerbsposition entscheiden, erfordern eine abteilungsübergreifende Gestaltung des gesamten Prozesses." ([BHT95], S.481)

Bei der Modellierung wird eine Top-down Sicht auf die Anwendungssituation eingenommen. Arbeitsgruppe und Arbeitsplatz leiten sich aus dieser Top-Down Sicht ab. Sie spielen bei dieser Sichtweise jedoch so eine untergeordnete Rolle, so daß ihre Analyse und Modellierung unproblematisch gesehen wird. Die Kooperationsformen können nach dieser Sichtweise rein nach funktionalen oder organisatorischen Gesichtspunkten beschrieben werden.

Die Modelle, die im Rahmen der Prozeßmodellierung erstellt werden, sollen

- die Prozesse und Funktionen der Prozesse zu erkennen und verstehen ermöglichen,
- bei der Analyse der Prozesse und Sichtbarmachen von Verbesserungsbedarfen unterstützen,
- Änderungen im Prozeßablauf sowie deren Auswirkungen zu erkennen ermöglichen [LHM95].

Um die Modellierung und Re(Engineering) von Unternehmensprozessen zu erleichtern, werden Referenzmodelle verwendet[NSZ95], [Krc97], [LHM95]. Dies wird wie folgt begründet:

"Die Erstellung von Prozeßmodellen für ein Unternehmen oder Teilbereiche eines Unternehmen erfordert viel Zeit und Know-how bezüglich der Modellierungsmethodik. Eine kostenmäßige Bewertung des dazu notwendigen Ressourceneinsatzes macht schnell deutlich, daß die Eigenerstellung der Prozeßmodelle gegenüber dem Zukauf fertiger Modelle unattraktiv erscheint." ([Krc97], S. 96)

Die Referenzmodelle stellen eine vorgedachte Lösung bzgl. einer optimierte Gestaltung von Geschäftsprozessen dar:

„Referenzmodelle sind Basismodelle, die aufgrund ihrer Wiederverwendbarkeit potentiell für die Erstellung mehrerer spezifischer Modelle herangezogen werden können. Dabei wird nicht gefordert, daß die Abbildungen der Realität in den Modellen vollständig sein muß; sie muß lediglich von Nutzen sein. Andererseits sollten Referenzmodelle so weit spezifiziert sein, daß sie auch ohne Veränderung sinnvoll als spezifische Modelle in einem Unternehmen verwendet werden können. Schließlich müssen Referenzmodelle an die spezifischen Gegebenheiten einer Organisation anpaßbar sein.“ ([NSZ95, S. 430)

Es wird bei dieser Sichtweise der Kooperationsmodellierung zwar erkannt, daß auch die sozialen Aspekte einer Organisation (Rolle des Menschen im Prozeß), und deren spezifische Kultur Auswirkungen auf die Möglichkeiten bei der Prozeßgestaltung haben können, jedoch werden bei dieser Sichtweise die Identifikation und (Neu)Gestaltung der Prozesse bei der Modellierung in den Vordergrund der Interesse gestellt.

Diese Schwäche der Business Process Reengineering wird in letzter Zeit kritisiert und die Wichtigkeit der Einbeziehung von der Gestaltung betroffenen anerkannt:

"BPR im herkömmlichen Sinn beschränkt sich in der Praxis meist auf die Dokumentation und Analyse komplexer, unternehmensweiter Prozesse. Zur Sicherung der Marktstellung des eigenen Unternehmens ist es jedoch unerlässlich, das Know-how aller Mitarbeiter über die betriebliche Abläufe zu erfassen und verfügbar zu machen." ([IDS99]

Im Gegensatz zum BPR-Konzept wird beim Organisationslernen das organisatorische Geschehen als fortlaufender Lernprozeß betrachtet, der alle Ebenen der Organisation in die Gestaltung

miteinbezieht. Die Entwicklung in Organisationen wird hierbei als eine permanenter Anforderung betrachtet [Rol97]. Unter Organisationslernen wird dabei,

" (...) der Prozeß der Erhöhung und Veränderung der organisationalen Wissensbasis, die Verbesserung der Problemlösungs- und Handlungskompetenz sowie die Veränderung des gemeinsamen Bezugsrahmens von und für Mitglieder innerhalb der Organisation zu verstehen." ([BP94], S.17)

Anforderung: Erstellung von Ist- und Sollmodellen

2.6.1.2 Arbeitsorientierte Sichtweise

Bei dieser Sichtweise sind bei der Modellierung die soziale Dimension der Arbeit, das Verstehen der Arbeitspraxis, die Interessen und Wünsche der Menschen im Mittelpunkt der Interesse gestellt:

"We started with the view that work is fundamentally social. Most activity, and certainly its meaning, arises in a context of cooperation. This led us to focus on the use situation as the basis for design." ([Kyn91], S. 65)

Es wird darauf hingewiesen, daß um die Nutzungssituation verstehen und angemessene Computerunterstützung gewährleisten zu können, neben der Kooperation auch andere Aspekte der Arbeit wie Macht, Konflikte und Kontrolle bei der Gestaltung beachtet werden müssen [Kyn91]. Die Partizipation der Beteiligten wird bei dieser Sichtweise besonders hervorgehoben. Somit bringen sie zum Ausdruck, daß die soziale Wirklichkeit nur aus der Erfahrung der beteiligten und betroffenen Individuen zu verstehen ist.

In diesem Zusammenhang wird die Modellierung als Abstraktion der Realität wegen ihrer Begrenztheit kritisiert:

"Our critique of modeling argues for the essential limitations of any and every form of representation. In other words, no representation, or set of representations is ever complete. Models are thus seen, in our view, as interpretations, as construction, which for some purposes, under certain conditions, used by certain people, in certain conditions may be found useful, not true or false." ([Ban95], S. 66)

Sie betonen jedoch auch, daß sie nicht gegen die Modellierung argumentieren, diese Kritik soll vielmehr zum Ausdruck bringen, daß das Bewußtsein bei der Modellierung geschafft werden muß, warum und was man modelliert. Aus diesem Grund schlagen sie vor, daß die Modellierung nicht als Abstraktion der Realität, sondern als "reframing" der Realität betrachtet werden sollte.

" (...) we recognize that representations are artifacts constructed from particular social locations and within specific forms of practice, we can expand our concern with the adequacy of representational forms to include ongoing dialogue and debate regarding the various places of representations in work and system design." ([Suc95a], S. 63)

Arbeitsorientierte Sichtweise legt auf die Analyse alltäglicher Arbeitspraxis, um verstehen zu können wie die Arbeit konkret ausgeführt wird. *"An activity-based view of work analyzes everyday work practices to demonstrate the ways employees actually make the business functions effectively."* ([Sac95], S. 38). Um diese detaillierte Verständnis von der Arbeit zu gewinnen, werden neben *„illustrations on paper, Foamcore models, computer-based prototypes, pictures, images, icons, mock-ups“* insbesondere ethnographische Methoden wie die Videoaufnahmen verwendet [Suc95a], [MW94]. Durch diese Art von Repräsentationen soll die Abstraktion der Realität überwunden bzw. gemildert und ein ganzheitliches Bild von der konkreten Arbeitssituation gemacht werden.

Als Vorteile der Videoaufnahmen werden folgende Punkte benannt:

- *"With video records, workers/users speak with their own voices. The portability of video representations make it possible für their voices to carry widely.*
- *Video openness to multiple viewing and interpretations enables unanticipated use.s.*
- *Video records make evidence for claims open to contest.*
- *Video records maintain the animation, dynamics of lived experience. (Working practice is lived experience, only partially representable) "* [Suc95a], S. 60]

| |
|---|
| Anforderung: Explizite Hinweis auf die Phänomene der Arbeitspraxis |
|---|

2.6.1.3 Formalisierungsorientierte Sichtweise

Diese Sichtweise der Kooperationsmodellierung geht davon aus, daß die anwendungs- und implementationsspezifische Aspekte durch die Netztheorie erfaßt werden können. Als wesentlicher Gründe für den Einsatz von Netztheorie werden folgende Punkte genannt:

"Netztheorie wird eingesetzt wegen ihrer Mächtigkeit und wegen ihrer guten Anpassung an das Vermögen von Menschen, bildliche Strukturen und Vorgänge zu erfassen und zu verarbeiten. Insbesondere bietet sie Vorteile bei der Darstellung nebenläufiger Handlungen und Funktionseinheiten und deren Strukturierung durch Vergrößerung über mehrere Abstraktionsschichten hinweg." ([JV87], S. 1f)

Die Benutzung von formalen Darstellungsmitteln wie z.B. Petri-Netze und die sie unterstützende Werkzeuge sollen die kompakte Darstellung komplexer Systeme und Abläufe auf unterschiedlichen Abstraktionsebenen ermöglichen [DO96]. Diese Art von Modellierung soll dann beim konzeptuellen Entwurf sowohl die semiformale Notation, als auch mathematisch exakte Sprache darstellen. Der Übergang zur vollständigen Beschreibung eines Systems geschieht durch systematische schrittweise Konkretisierung und Vervollständigung der mathematischen Objekte. Als wichtigste Gründe für die Formalisierung bezogen auf die Petri-Netze werden folgende Gesichtspunkte genannt:

" Ihre graphische Notation erlaubt eine intuitive anwendernabe Ebene, (...) und schließlich lassen sich Petri-Netze ausführbare Sprache interpretieren. Darüber hinaus sind Petri-Netze eine ideale formale Grundlage anderer Sprachen und Notationen; sie erlauben die semantische Vergleichbarkeit unterschiedlicher Notationen, und Übergänge zwischen Sprachen lassen sich als Transformationen der entsprechenden Petri-Netze formalisieren." ([DO96], S. 360)

Um die anwendungsspezifischen Aspekte zu erfassen wird vorgeschlagen, Petri-Netz-artige Darstellungen mit speziellen, anwendungsnahen graphischen Symbolen zu verwenden. So z.B. die Objekte und Aktivitäten können durch spezielle Symbole (Pictogramme, Icons) repräsentiert, so daß sie auch als Basis für die Kommunikation mit der Anwenderseite verwendet werden können [Obe87].

Charakteristisch für diese Sichtweise ist, daß die Techniken "Verfeinerung" und "Vergrößerung" bei der Modellierung einen besonderen Stellenwert haben. Bei der Verfeinerung wird davon ausgegangen, daß die komplexen Systeme sich nicht in einem "Wurf", sondern nur in einem schritt-

weise verfeinerndem Entwicklungsprozeß konzipieren lassen. Das Gesamtsystem kann hiermit in detailliertere (Sub-) Systeme gegliedert werden.

Die Technik "Vergrößerung" geht umgekehrt vor als die Technik der Verfeinerung. Es kann ausgehend der Analyse von Subsystemen, schrittweise zur Entwurf des Gesamtsystems übergegangen werden.

| |
|---|
| Anforderung: Implementierbarkeit der Modelle |
|---|

2.6.1.4 Pragmatikorientierte Sichtweise

Es wird in der Literatur und in der Praxis zunehmend auf die Probleme der praktischen Entwicklungsarbeit in Softwareprojekten hingewiesen. Eine Studie, die im Rahmen des Forschungsverbundes Software-Engineering durchgeführt worden ist, stellt als Ergebnis folgendes heraus:

"Es mangelt an Durchgängigkeit des Entwicklungsprozesses, d.h. der methodischen und technischen Unterstützung konsistenter Übergänge zwischen den einzelnen Entwicklungsschritten. Dies muß durch aufeinander angepaßte Beschreibungstechniken und abgestimmte Werkzeuge behoben werden" [BUB99], S. 34)".

Auf der anderen Seite wird auch die These vertreten,

"(...), daß die Entwicklung von allumfassenden Methoden, Modellierungssprachen oder gar Referenzmodellen kaum realistisch ist. Diesbezüglich fällt auf, daß in der Literatur vorgeschlagenen Modellierungstechniken meist zu kompliziert und theoretisch sind; sie dienen oft eher als "Baukasten", aus dem die für die spezielle Unternehmenssituation relevanten - und leicht verständlichen - Modellierungselemente gewählt werden." ([JKK99], S.87)

Pragmatikorientierte Sichtweise hat von daher das Ziel, die praktische Entwicklungsarbeit in Softwareprojekten zu unterstützen. Es wird in diesem Zusammenhang insbesondere auf die Wichtigkeit der Flexibilität der Modellierungsmethoden und Werkzeugen hingewiesen. Aufgrund der Tatsache, daß die Rahmenbedingungen und Ziele je nach Organisation sich voneinander unterscheiden, wird gefordert, daß die Methoden und Werkzeuge in bezug auf die Modellierung "customizeable" sein sollten [BOC99]. Hiermit stellt diese Sichtweise die Anpaßbarkeit, Erwei-

terbarkeit der Modellierungsmethoden an geänderte Rahmenbedingungen und Ziele im Vordergrund der Interesse [JKK99].

Es wird versucht, die Integration fachlicher Modelle (wie z.B. Geschäftsprozesse) mit den IT-Modellen durch Werkzeuge herzustellen, so daß eine große Flexibilität bei der Modellierung erreicht werden kann. Bei dieser Sichtweise steht die möglichst schnelle Umsetzung der Modellierungsergebnisse in die Praxis im Vordergrund der Interesse. Zu diesem Zweck werden z.B. die Schnittstellen der Modellierungswerkzeuge offengelegt, so daß die Kopplung mit anderen Tools reibungslos und schnell zum Zwecke der Umsetzung erfolgen soll.

"IBM und BOC eröffnen ihren Kunden durch die Kopplung des Geschäftsprozeßmanagement - Werkzeuges ADONIS® (BOC) mit dem Workflow-Management- System MQSeries Workflow™ (IBM) neue Möglichkeiten für die Übernahme von Geschäftsprozeßmodellen aus der Analysephase in ausführbare Production Workflows und die Auswertung ausgeführter Workflows." [BOC99]

Hierbei wird versucht, eine geschäftsprozeßgetriebene Anwendungsentwicklung zu ermöglichen, bei der ausgehend von fachlichen Geschäftsprozeßmodellen die Verbindung zur Anwendungsentwicklungsumgebungen hergestellt werden kann.

| |
|---|
| Anforderung: Werkzeugunterstützung |
|---|

Die, in dieser Teil der Arbeit herausgearbeiteten Anforderungen werden im fünften Kapitel zur Bewertung der Ansätze herangezogen. Im folgenden wird der analytische Vergleich der Ansätze durchgeführt.

Kapitel 3

Ansätze zur Kooperationsmodellierung - ein vergleichender Überblick

3.1 Auswahl der Methoden /Systematik des Vergleichs

In diesem Kapitel werden vier Ansätze zur Kooperationsmodellierung analytisch verglichen. Da es eine Vielzahl verschiedene Methoden gibt, mußte zwangsläufig eine Auswahl getroffen werden. Obwohl die objektorientierten Methoden, insbesondere die Modellierungssprache UML (Unified Modeling Language) für die SoftwareentwicklerInnen in der Praxis bei der Analyse, Design und Implementierung eine große Hilfestellung bietet, ist sie nicht für den Vergleich ausgewählt worden. Der wesentliche Grund war, daß die UML kein explizites Konzept von Kooperation hat und von daher die Aspekte von Kooperation mit dieser Methode nicht explizit thematisiert werden können. Dies ist jedoch, wenn die Definition von "Kooperationsmodellierung" betrachtet wird, die Voraussetzung für die Auswahl gewesen. Der zunächst für den Vergleich ausgewählter Ansatz "ADONIS" wurde auch aus diesem Grund, aus der Liste von zu vergleichenden Ansätzen gestrichen. Die Kooperationsbilder sind ebenfalls - obwohl sie sehr gut die kooperativen Aufgaben der beteiligten Akteure, Artefakte, die Art und Weise der Weitergabe von Informationen und Arbeitsgegenständen veranschaulichen - nicht für den Vergleich ausgewählt. Der wesentliche Grund hierfür war, daß für die Kooperationsbilder kein Metamodell zur Verfügung steht.

Die Kriterien für die Auswahl von Ansätzen können wie folgt zusammengefaßt werden:

1. *Explizite Konzepte von Kooperation*, damit im Modell Aspekte von Kooperation explizit gemacht werden können;

2. *Graphische Modellierung*, damit die Modelle als Diskussions- und Kommunikationsgrundlage herangezogen werden können;
3. *Metamodell* (ein Modell, das die Konzepte einer Modellierungstechnik, die verwendbaren Modellelemente und deren Zusammenhänge modelliert) Sie ist notwendig, um die Konzepte darzulegen, um die Methode besser verstehen zu können und um einen Konsens über ihre Verwendung und Bedeutung herbeizuführen.

Es sind die folgenden Ansätze zur Kooperationsmodellierung für den Vergleich ausgewählt, die alle oben genannten Kriterien erfüllen:

1. **RFA-Netze:** Rollen-/Funktions-/Aktions-Netze [Oberquelle]
2. **SeeMe:** Sozio-orientierte semistrukturierte Modellierungsmethode [Herrmann et al.]
3. **ARIS:** Architektur integrierter Informationssysteme [Scheer]
4. **Action/Workflow** [Winograd/Flores]

Bei der Auswahl dieser Ansätze - neben der Erfüllung der oben genannten Kriterien - war auch wichtig, daß sie sich voneinander konzeptionell unterscheiden. Die Rollen-/Funktions-/Aktionsnetze wurden insbesondere aufgrund ihrer Fundierung auf die Petri-Netze - die jedoch um die Konzepte von Kooperation erweitert worden sind - ausgewählt. Da in der Praxis und in der Wissenschaft viele Ansätze (z.B. FUNSOFT-Netze) auf die Netztheorie basieren, schien dieser Auswahl angemessen. SeeMe wurde insbesondere wegen dem Konzept zur "vagen Modellierung" ausgewählt, wodurch sie sich von den anderen drei Ansätzen unterscheidet. Bei der Auswahl von ARIS standen, einerseits der Bekanntheitsgrad der Methode "erweiterten ereignisgesteuerten Prozeßketten und andererseits ihre Orientierung auf die Geschäftsprozeßmodellierung, in den Vordergrund. . Bei der Auswahl des Ansatzes "Action/Workflow spielte ihre theoretische Fundierung auf die Sprechakttheorie im Vordergrund der Interesse.

Es sei an dieser Stelle der Arbeit darauf hingewiesen, daß die Anzahl der zugrundegelegten Informationsmaterialien sich wesentlich voneinander unterscheiden. So z.B. standen für ARIS drei Bücher, eine Vielzahl von Artikel in Zeitschriften und aktuelle Informationen aus der WWW zur Verfügung. Im Gegensatz dazu stand für die RFA-Netze lediglich die Habilitationsschrift [Obe87] zur Verfügung. Außerdem ist weiterhin zu bemerken, daß die Entstehungsjahre der Ansätze sich auch wesentlich voneinander unterscheiden. So z.B. stammen die RFA-Netze aus dem Jahre 1986 und die Modellierungsmethode SeeMe aus dem Jahre 1997.

Die ausgewählten Ansätze werden im folgenden nach einem vordefinierten Analyseraster verglichen. Es werden dabei folgende Kategorien der Analyse unterschieden:

1. Entstehungsgeschichte,
2. Theoretische Fundierung
3. Basiselemente und ihre Verknüpfung
4. Einsatzkontext

Bei der Definition dieses Analyserasters ist insbesondere darauf geachtet, daß sie einen Beitrag zum Verständnis der Ansätze beitragen.

3.2 Ansätze zur Kooperationsmodellierung

3.2.1 Rollen-/Funktions-/Aktions-Netze

Rollen/Funktions-/Aktions-Netze (RFA-Netze) sind im Rahmen der Habilitationsschrift von Prof. Dr. Oberquelle, mit dem Titel „Sprachkonzepte für die kooperative Rollenentwicklung“, entstanden [Obe87]. Sie wurde im Jahre 1986 vom Fachbereich Informatik der Universität Hamburg angenommen. Insbesondere die Arbeiten im Arbeitsbereich "Theoretische Grundlagen der Informatik" stellten für die Entwicklung dieser Netze eine inspirierende Umgebung. In diesem Ansatz wird die Arbeit aus der Perspektive einzelner Personen betrachtet. Die Gesamttätigkeit einer Person wird dabei als Handeln in verschiedenen Rollen, die in ein Netzwerk kooperierender Rollen eingebettet sind, aufgefaßt. Es werden insbesondere die Rollen und ihr Zusammenwirken hervorgehoben und auf drei Ebenen betrachtet: Rollenebene, Funktionsebene und Aktionsebene.

Prof. Dr. Oberquelle beschäftigt sich zur Zeit am Fachbereich Informatik der Universität Hamburg im Arbeitsbereich "Angewandte und Sozialorientierte Informatik" (ASI), hauptsächlich mit Software-Ergonomie und CSCW. Es ist mir leider nicht bekannt, ob diese Netze erweitert und auf der Basis dieser Netze Systeme implementiert worden sind. Ich gehe daher in dieser Arbeit davon aus, daß dieser Ansatz in der Praxis nicht weit verbreitet ist. In der Forschung findet dieser Ansatz eine breitere Verbreitung als in der Praxis [Obe96], [HHL98].

Entstehungsgeschichte

Ausgangspunkte für die Entwicklung von Rollen-/Funktions-/Aktions-Netzen (RFA-Netze) waren damaliger Schwierigkeiten, die sich im Bereich der Softwaretechnik und bei der Benutzung

interaktiver Programmsysteme zeigten: „Sowohl für die Kommunikation von Entwicklern mit Benutzern, wie für die Formulierung von Benutzungsmodellen fehlen adäquate Beschreibungssprachen.“ ([Obe87], S. 289). Oberquelle stellt dabei insbesondere die mangelnde Rückkopplung zu den Anwendern bei der Softwareentwicklung sowie die Art und Weise der Modellierung und Beschreibung, die durch computerorientierte Grundvorstellungen geprägt sind, als Problemfelder der Softwareentwicklung, hervor. Er sieht dabei die Notwendigkeit für die Entwicklung einer Sprache, die einerseits die Kooperation zwischen Entwickler und Benutzer ermöglicht und andererseits zur transparenten Beschreibung der Computersysteme und der Entwurf von Mensch-Maschine-Schnittstellen beiträgt. Diese Erkenntnisse führten ihn dann zur Entwicklung von RFA-Netzen, die einen Beitrag zur Entwicklung einer benutzerorientierten Beschreibungssprache für Organisationen mit interaktiver Computeranwendung leisten sollte. Sie sollte die Möglichkeit bieten, die abstrakte Gesamtarchitektur von Anwendungssystemen, einschließlich Benutzungsmodellen von computergestützten Teilen aus der Sicht von Benutzern zu erfassen und transparenter zu machen.

Theoretische Fundierung

Rollen/Funktions-/Aktions-Netze bauen hauptsächlich auf die Netztheorie auf, die um spezielle Symbole erweitert worden sind. Die Netztheorie ist eine Theorie der Systemorganisation, die im Jahr 1962 mit der Dissertation von C. A. Petri begründet wurde. Das Ziel der Netztheorie besteht im wesentlichen darin, möglichst viele Erscheinungen bei der Informationsübertragung und bei der Informationsverarbeitung in einheitlicher und exakter Weise beschreiben zu können. Als Ausdrucksmittel werden dabei Netze verwendet, die gleichzeitig auch als Objekt einer mathematisch präzisen Behandlung und als Mittel einer graphischen Veranschaulichung dienen ([Lex97], S.577).

Die spezielle Netztheorie befaßt sich mit dem Fluß der Ressourcen (z.B. Objekte) durch ein Netz und untersucht die Art, wie die Flüsse koordiniert und verzweigt oder zusammengeführt werden. Mit bestimmten Methoden können Probleme der Lebendigkeit und Sicherheit von Systemen behandelt werden. Auf einer niedrigen, detaillierten Stufe ist die Modellierung von Systemen, die Analyse ihres Verhaltens und die Verifikation von Eigenschaften möglich ([Lex97], S.578).

Petri-Netze werden seit über 30 Jahren zur Modellierung von Systemen und Systemabläufen eingesetzt. Im Laufe dieser Zeit wurde die Theorie zu den Netzen und die Netze selbst weiter untersucht und entwickelt. Diese Netze sind graphische Formalismen zur Ablaufbeschreibung. Sie ermöglichen die Beschreibung sequentiellen, sich gegenseitig ausschließenden sowie nebenläufigen

gen Aktivitäten. Dabei werden nur wenige graphische Beschreibungskonstrukte benötigt, nämlich Kreise (zur Repräsentation statischer Aspekte, z.B. Dokumente, Ressourcen, Daten usw.) und Vierecke (zur Repräsentation von Ereignissen, Aktivitäten oder lokalen Zustandsübergängen) [Obe96].

Basiselemente und ihre Verknüpfung

Die Modellierung in diesem Ansatz erfolgt auf drei Betrachtungsebenen. Die statische Struktur von Rollen- und Funktionsgefügen wird auf der Rollen- und Funktionsebene beschrieben. Die Dynamik hingegen wird auf der Aktionsebene, mit Hilfe von beschrifteten dargestellt. Durch die statische Beschreibung von Rollen und Funktionen können Formen der Arbeitsorganisation auf der Rollen- und Funktionsebene erfaßt werden. Zur Darstellung der statischen Struktur auf dieser Ebene können verschiedene Netzarten, die eine Unterklasse von RFA-Netzen bilden, verwendet werden: die Rollennetze (R-Netze), Funktionsnetze (F-Netze) und Rollen- und Funktionsnetze (RF-Netze).

Die dynamische Struktur der Arbeitsabläufe kann auf der Aktionsebene mit Aktionsnetzen detailliert beschrieben werden. Auf dieser Ebene können zum einen die Kontrolle von Handlungen durch Funktionsträger (Kontrollfluß) und zum anderen die Bearbeitung von Objekten in Handlungen (Objekt- bzw. Datenfluß) getrennt modelliert und später in Aktionsnetzen überführt werden.

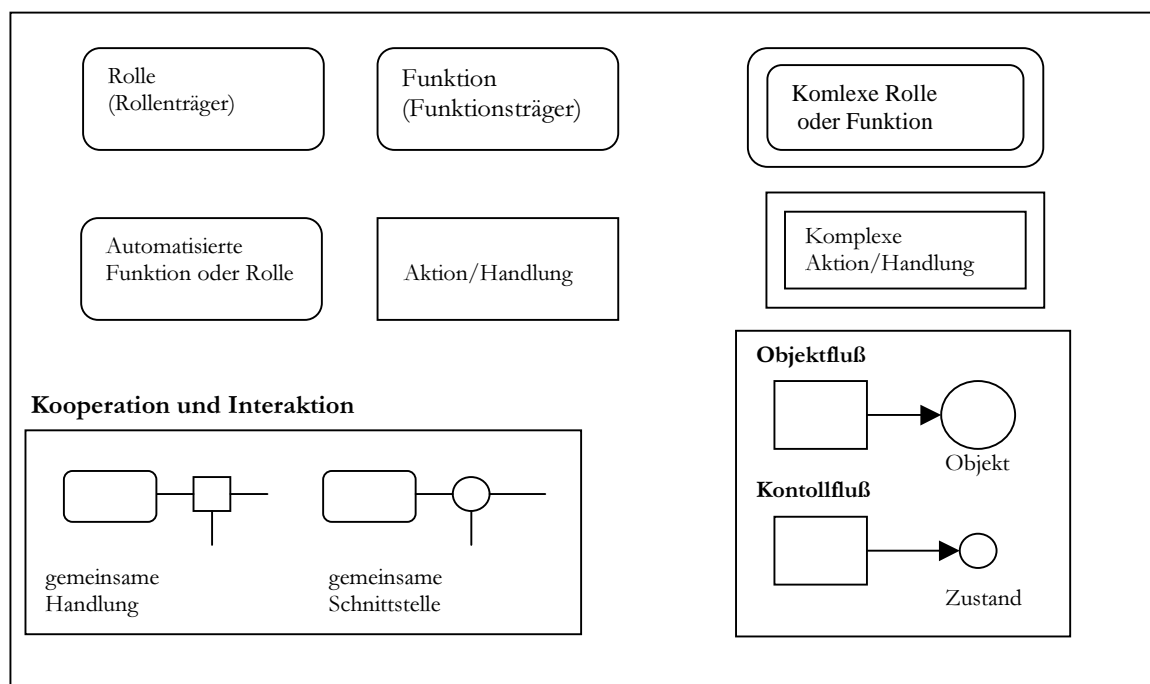


Abbildung 4: Ausgewählte Elemente der RFA-Notation

Rolle: Unter einer Rolle werden in RFA-Netzen die Aufgaben, die durch Arbeitsteilung mit einheitlichem Ziel entstanden sind, zusammen mit dem zu ihrer Erfüllung notwendigem Wissen und allen Hilfsmitteln, zusammengefaßt. Für diese Aufgabe verantwortliche Person wird als Rollenträger bezeichnet. Einheiten können dabei als Rollen identifiziert werden, wenn sie Tätigkeiten, Handlungen oder Funktionen ausführen.

Funktion: Unter Funktionen werden Tätigkeiten, ihre Träger und die benötigte Mittel zusammengefaßt. Dabei werden als Tätigkeiten - die zusammenhängende Gesamtheiten von sequentiellen Handlungen einer Rolle mit gemeinsamem Zweck und bestimmte Methode als Einheit - bezeichnet. Tätigkeiten tragen zur Aufgabenerfüllung der Rolle bei und können von Personen oder Maschinen ausgeführt werden.

Rollen und Funktionen werden durch gleichartige Symbole dargestellt jedoch durch eindeutige Beschriftung unterschieden. Rollen werden durch personenorientierte Namen (z.B. Leser, Leiter usw.), Funktionen durch substantivierte Verben, die primär die Tätigkeit ausdrücken (z.B. Beschaffung) bezeichnet. Die jeweiligen Träger können ergänzend in Klammern aufgeführt werden.

Zur Hervorhebung von automatisierten Funktionen und teilweise automatisierten Rollen kann die untere Kante der zugehörigen Kästen jeweils verstärkt gezeichnet werden. Rollen- und Funktionen, die an anderer Stelle verfeinert werden, können mit doppelten Begrenzungslinien gezeichnet und in der Verfeinerung mit den gestrichelten Linien kenntlich gemacht werden. Die Kennzeichnung von komplexen Netzkomponenten erfolgt durch doppelte Umrandungen. Die hierarchische Anordnung von Rollen und Funktionen kann durch Schachtelung der zugehörigen Kästen ausgedrückt werden. An dieser Stelle ist darauf hinzuweisen, daß in RFA-Netzen die Funktionen keine Rollen beinhalten dürfen. Das umgekehrte ist jedoch erlaubt.

Kooperations- bzw. Interaktionsbeziehungen: In RFA-Netzen werden unterschiedliche Rollen soweit sie nicht durch explizite Kooperationsbeziehungen miteinander verbunden sind, unabhängig voneinander betrachtet. Wesentlicher Unterschied zwischen Kooperation und Interaktion besteht darin, daß bei der Interaktion die soziale Dimension (Abstraktion von der Rollenzugehörigkeit von Funktionen und von Rollenkontexten) einer Rolle vernachlässigt wird. Es können zwei Arten von Kooperation bzw. Interaktion getrennt graphisch modelliert werden. Zum einen die Kooperation durch gemeinsame Handlungen (z.B. persönliche Übergabe eines Doku-

menten, Austausch von Informationen im Gespräch) und zum anderen die Kooperation durch die Benutzung gemeinsamer Schnittstellen (z.B. Austausch von Objekten zwischen Personen).

Mit Aktionsnetzen können die Arbeitsabläufe d.h. das Zusammenspiel von Funktionsträgern, Handlungen und Objekten in Zeit und Raum beschrieben werden. Kontroll- und Objekt netze als Teilklassen von Aktionsnetzen ermöglichen zwei verschiedene Aspekte getrennt zu modellieren. Der aktuelle Zustand der Funktionsträger in bezug auf eine einzelne Funktion kann in diesen Netzen durch Markierung angegeben werden. Es wird an dieser Stelle die in [Obe87] ausführlich beschriebenen Markierungsregeln und andere Details nicht eingegangen.

In diesen Netzen werden die *Handlungen/Aktionen* als Bestandteile von Tätigkeiten aufgefaßt. Wenn die Handlungen sich in kleinere Einheiten aufspalten lassen, werden sie als komplex bezeichnet und durch doppelte Umrandungen gekennzeichnet. Die elementaren Handlungen/Aktionen werden durch Kästchen mit entsprechender Bezeichnung dargestellt. Auf dieser Ebene wird dem Objekt konzept eine besondere Aufmerksamkeit gewidmet. Objekte stellen in RFA-Netzen die Verbindung zwischen statischen und dynamischen Sicht dar. Handlungen werden in RFA-Netzen immer objektorientiert betrachtet. Objekte können dabei auf verschiedene Art und Weise dargestellt (Textdarstellungen, graphische und bildliche Darstellungen) werden - zum Objektbegriff und Darstellungsarten siehe [Obe87], ab S.129).

Einsatzkontext

RFA-Netze können in der Wissenschaft und dort eingesetzt werden, wo auch Petri-Netze eine Anwendung finden. Die Bereiche, in denen die Petri-Netze eingesetzt werden, reichen von dem Hardware-Entwurf, Rechner Organisation bis zur Geschäftsprozeß-, Workflow- und Unternehmensmodellierung.

Diese Netze können weiterhin zur Modellierung von Dialogsystemen und zur Entwurf von Mensch-Maschine-Schnittstellen herangezogen werden. Die Ablaufbeschreibung, die in Aktionsnetzen modelliert wird, kann zur Generierung von Programmen verwendet werden. Da mit diesem Ansatz modellierte Sachverhalte sich hauptsächlich auf die Darstellung der Austausch von Objekten bezieht, ist er insbesondere in den Situationen, wo dieser Aspekt von Kooperation, im Vordergrund steht zu empfehlen. RFA-Netze sind wegen ihrer graphischen Differenziertheit insbesondere für die Beschreibung komplizierter Zusammenhänge geeignet. Eine Präzisierung bis

hin zur Formalisierung der RFA-Konzepte ist für Aktionsnetze ohne großen Schwierigkeiten möglich [Obe97].

3.2.2 SeeMe

Die Modellierungsmethode SeeMe (*sozio-orientierte semistrukturierte Modellierungsmethode*) ist an der Universität Dortmund am Fachgebiet Informatik und Gesellschaft von Prof. Thomas Herrmann, Marcel Hoffmann und Kai-Uwe Loser im Jahre 1997 entwickelt worden. Es wird zur Zeit an der Entwicklung eines Editors namens EasySeeMe für die SeeMe Modellierungsmethode gearbeitet. Kennzeichnend für diesen Ansatz der Kooperationsmodellierung ist, daß die Methode die Modellierung von "vagen Informationen" ermöglicht. Sie wird insbesondere an der Universität Dortmund, wo auch SeeMe entstanden ist, in Lehrveranstaltungen eingesetzt. Weiterhin wurde sie in zwei kommerziellen Unternehmen [Her99a] und zwei Forschungsprojekten MOVE [HSW98] und LOOK [Loo99] eingesetzt

5.2.1. Entstehungsgeschichte

Die Entwicklung der Modellierungsmethode SeeMe geht hauptsächlich auf Anforderungen an Modellierungsmethoden aus dem Requirements-Engineering und der Beschreibung sozio-technischer Systeme sowie auf einer Analyse einiger bestehenden Modellierungsmethoden zurück. Zu den analysierten Methoden gehörten: objektorientierte Methode Unified Modeling Language (UML), die erweiterten ereignisgesteuerten Prozeßketten (eEPKs) aus ARIS, Erweiterungen von Entity-Relationship-Modellen (eERM), Notationen zur Verhaltensmodellierung wie Statecharts und Petri-Netz basierte Ansätze wie FUNSOFT-Netze und RFA-Netze [Herr99a], [HHL98]. Sie identifizierten dabei folgende Schwachpunkte der analysierten Methoden:

- Die Erzwingung von Vollständigkeit bei der Modellierung. Die Möglichkeiten, vage bzw. unvollständige Informationen zu kennzeichnen ist begrenzt.
- Angebot von "künstlichen Konzepten", die zwar den Entwurf von Software unterstützen, jedoch Anwendern und Benutzern schwer vermittelbar sind.
- Mangelnde Modularisierung und Anpaßbarkeit der Methoden an unterschiedliche Anwendungsfälle.
- Hinderliche Einschränkungen bei der Modellierungsfreiheit.
- Die Möglichkeiten, Attribute einzuführen und darzustellen, sind begrenzt.

Um diese genannten Lücken zu schließen, sahen sie die Notwendigkeit der Entwicklung einer neuen Methode:

„Hintergrund der Entwicklung einer neuen Modellierungsmethode ist die bessere Unterstützung bei der Erstellung von z.B. mitarbeiterorientierten Anforderungsanalysen oder bei der sozial-verträglichen Einführung von Informations- und Kommunikationssystemen sowie bei der ergonomischen Bewertung von Software. Datenflußdiagramme, der Entity-Relationshipansatz oder moderne Methoden wie ARIS/UML zeigen meistens negative Konsequenzen bezüglich der Sozialverträglichkeit. Genau diese Lücke soll SeeMe (sozio-orientierte semistrukturierte Modellierungsmethode) schließen.“ ([Her99], S.125)

Theoretische Fundierung

Die Methode SeeMe besitzt von Haus aus keine theoretische Begründung. Sie ist wie oben schon erwähnt, aufgrund der festgestellten Schwächen der vorhandenen Modellierungsmethoden entstanden. Es wird insbesondere auf die Konzepte der untersuchten Methoden wie ereignisgesteuerten Prozeßketten, der objektorientierten Modellierung und State-Charts nach Harel Bezug genommen.

In diesem Zusammenhang ist weiterhin zu erwähnen, daß Goguens Arbeit [Gog94] für die Entwicklung des Konzepts „vages Modellieren“ eine besondere Rolle gespielt hat. Basierend auf die Arbeit von Goguen, sahen die Entwickler von SeeMe-Methode die Notwendigkeit bei der Anforderungsanalyse, die Kombination unterschiedlich feuchter bzw. trockener Informationen in einem Modell zu repräsentieren und zu kombinieren ("Metapher der feuchten Informationslandschaft"). Die Information, die auf der Basis eines weiten Spektrums unterschiedlicher Arten von Kontext gewonnen werden können, werden dabei als "trockener Information" bezeichnet. Nasser Information dagegen läßt sich nur in Bezug zu einem sehr speziellen, situationsabhängigen Kontext gewinnen. Weiterhin wird die Information, die zwischen diesen beiden Extremen liegt, als feuchte Information bezeichnet.

In diesem Sinne kann SeeMe als eine neue Methode, die einerseits die Vorteile der bestehenden Methoden verbindet und andererseits neue Konzepte wie "vages modellieren" zum Zwecke der Darstellung feuchter Informationslandschaften anbietet, betrachtet werden.

Basiselemente und ihre Verknüpfung

Die Modellierungsmethode SeeMe enthält als Basiselemente Rollen, Aktivitäten, Entitäten und als Erweiterung Darstellungsmitteln für vages Modellieren. In SeeMe hat jedes Basiselement als notwendige Attribute einen Namen oder Kurzbeschreibung. In SeeMe wird nicht zwischen statischen und dynamischen Aspekten unterschieden. Beide Aspekte können in einem Modell dargestellt werden. Im folgenden werden die Elemente der SeeMe zusammenfassend dargestellt. Anschließend werden sie näher aufgeführt:

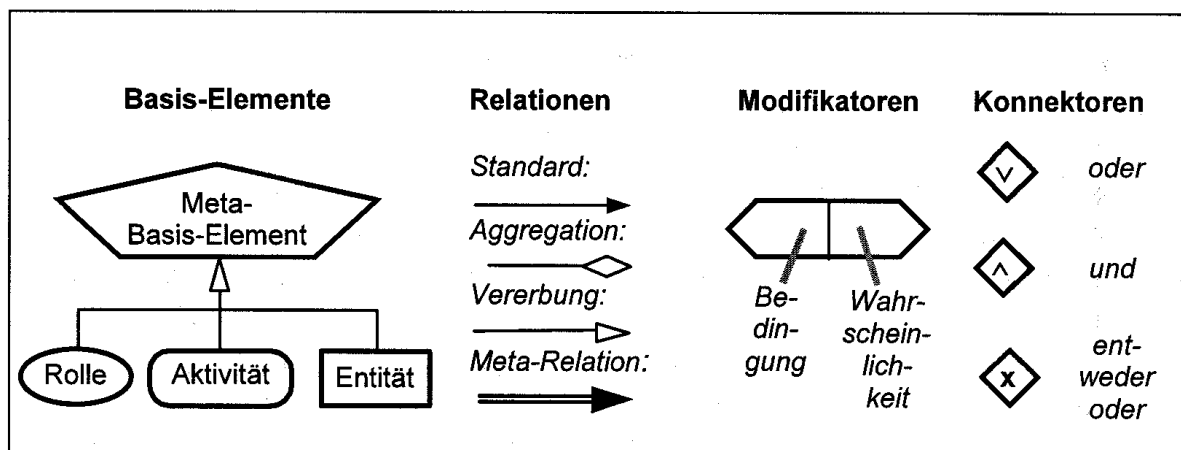


Abbildung 5: Elemente der SeeMe-Notation [HHL99]

Rolle: Eine Rolle repräsentiert in SeeMe eine Menge von Rechten und Pflichten, die z.B. einer Person, einer Abteilung, einer Arbeitsgruppe oder einer anderen organisatorischen Einheit zugeordnet sind. Eine Rolle kann eine Person oder aber auch eine Funktion einer Person repräsentieren. Dementsprechend kann eine Person oder eine Gruppe von Personen mehrere Rollen ausfüllen. Rollen werden als Kreis oder Ellipse dargestellt.

Aktivitäten: Aktivitäten beschreiben Verhalten. Aktivitäten haben Start- und Endzeitpunkte und können zeitlich zueinander ins Verhältnis gesetzt werden. Sie repräsentieren Operationen an Entitäten, Arbeitsaufgaben, Verrichtungen und Tätigkeiten, die von Rollen ausgeführt werden.

Entitäten: Entitäten sind Ressourcen zur Ausführung von Aktivitäten. Sie werden von Aktivitäten genutzt oder verändert. Entitäten repräsentieren Dokumente, Dateien, Nachrichten, Wissen

oder Information, die von einer Aktivität an eine andere gegeben werden. Weiterhin repräsentieren sie die Arbeitsmittel, die Rollen bei der Ausführung ihrer Aktivitäten unterstützen. Entitäten werden durch Rechtecke dargestellt.

Relationen zwischen Basiselementen: Eine Beziehung zwischen zwei Elementen wird in SeeMe durch Relationen dargestellt. Es kann dabei jedes Basiselement zu einem anderen Basiselement in Beziehung gesetzt werden. Das Aufbauen einer solchen Beziehung wird als Instantiierung oder in Kraft treten einer Relation bezeichnet, mit dem auch ein Ereignis verbunden ist. Relationen zwischen Basiselementen können beliebig benannt werden. Sie können auch von beliebigem Typ sein. Relationen zwischen Basiselementen werden durch gerichtete Kanten dargestellt. Im folgenden werden Standardbedeutungen der Relationen aufgeführt, die gültig sind, wenn der Typ einer Relation zwischen zwei Basiselementen nicht explizit anders benannt wird.

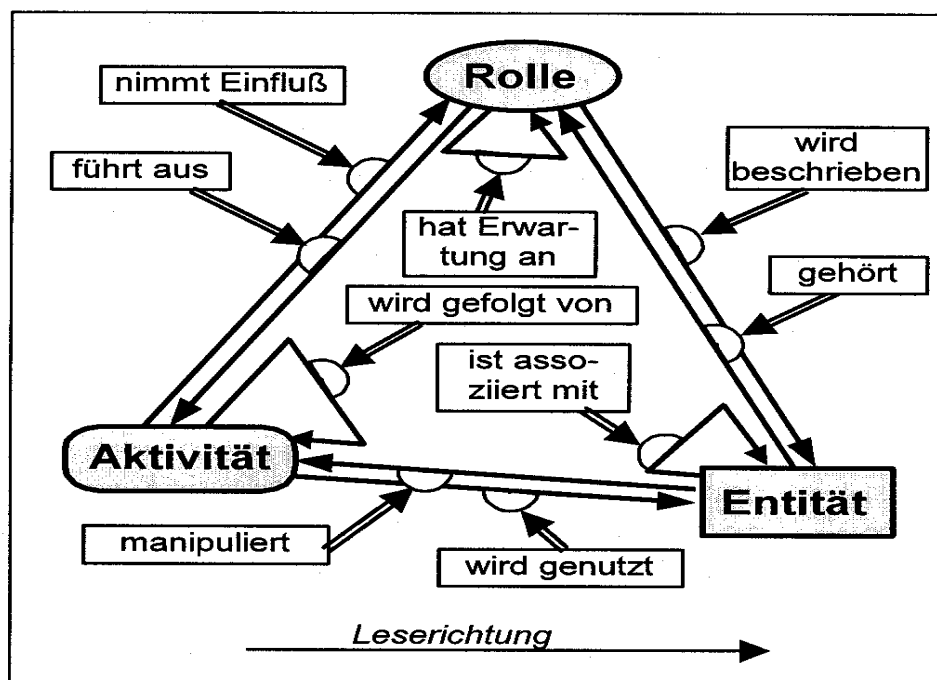


Abbildung 6: Vorgegebene (Standard)Bedeutung von Relationen zwischen Basiselementen

Konnektoren: Logische Verknüpfungen von Relationen werden mit Hilfe von Konnektoren beschrieben. Das SeeMe Notationssymbol für die Konnektoren ist um ein 45Grad gedrehtes Quadrat. Grundlegende Konnektoren in SeeMe sind UND und XOR (entweder oder). Weitere

Konnektoren wie z.B. das nicht-exclusive ODER, können durch Kombinationen definiert werden. Konnektoren können zwei oder mehrere Relationen miteinander verknüpfen, so daß eine neue Relation entsteht.

Bedingungen: SeeMe enthält für Bedingungen das Konzept Modifikators. Modifikatoren können auf alle Relationen und auf Basiselemente angewendet werden. Das Notationssymbol für Modifikatoren ist das gedehnte Sechseck. Modifikatoren können in SeeMe zwei Teile beinhalten, die durch einen senkrechten Strich getrennt sind. Im ersten Teil können Ereignisse, Zustände oder logische Bedingungen angegeben werden, die gegeben sein müssen, damit die modifizierte Relation in Kraft tritt. Im zweiten Teil kann angegeben werden, mit welcher Unsicherheit bzw. Wahrscheinlichkeit die Bedingung eintritt. Die Wahrscheinlichkeit kann entweder quantitativ durch die Wahl eines Wahrscheinlichkeitswertes zwischen 0 und 1 oder qualitativ durch eine Häufigkeitsangabe (ausnahmsweise, gelegentlich, oft ...) oder eine deontologischer Spezifizierung (erlaubt, wünschenswert, verboten ...) ausgedrückt werden.

Einbettung: Mit Hilfe des Konzepts der Einbettung können in SeeMe unterschiedliche Abstraktionsmechanismen zwischen Unter- und Oberelementen ausgedrückt werden.

Es besteht in SeeMe weiterhin die Möglichkeit Aktivitäten näher zu beschreiben. Dies wird entweder durch Hinzufügen von Attributen oder durch die Detaillierung von Aktivitäten durch Subelemente erreicht.

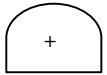
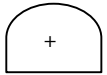

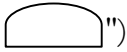
Da das Konzept des vagen Modellierens kennzeichnend für die Modellierungsmethode SeeMe steht, womit sie sich von anderen Ansätzen unterscheidet, wird im folgenden auf dieses Konzept näher eingegangen.


Vages Modellieren

Vages Modellieren dient in SeeMe zur Erweiterung der vorgestellten Basiselemente und zur Darstellung von feuchten Informationen. SeeMe enthält zwei Konzepte zur vagen Modellierung. Sie sollen zur Vermeidung von zu starker Formalisierung beitragen.

1. Beabsichtigte Auslassung von Information

Beabsichtigte Auslassung von Information soll die Übersichtlichkeit, Verständlichkeit und Deutlichkeit durch das Verstecken weniger wichtiger Darstellungen erhöhen. SeeMe sieht zu diesem Zweck drei Alternativen vor:

- Verweise auf ergänzende Informationen, die beim Modellierer abgerufen werden können, werden an Basiselementen durch eine leere Fläche in Form eines Halbkreises mit einem Plusymbol notiert () ("  ").
- Verweise auf ergänzende Informationen in anderen aktuell nicht sichtbaren Modellteilen werden an Basiselementen durch schwarze Flächen notiert ("").
- Bewußt in Kauf genommene Modellierungslücken, deren Schließung nicht als notwendig erachtet wird, werden durch eine leere Fläche, z.B. in Form eines Halbkreises notiert ("")

Um Mißverständnisse in bezug auf die Vollständigkeit zu vermeiden, existiert in SeeMe eine Notation für die Kennzeichnung vollständiger Information. Hierzu kann einen Haken () benutzt werden, indem er an Teile von Modellen annotiert wird.

2. Kennzeichnung unsicherer Information

Mit diesem Konzept bekommt der Modellierer die Möglichkeit, darauf hinzuweisen, daß die zugrunde liegenden Informationen unvollständig oder unsicher sind. Hierbei werden ebenfalls drei Fälle unterschieden:

- Wenn der Modellierer feststellt, daß eine Spezifikation unvollständig ist und er auch nicht in der Lage ist, sie zu vervollständigen, kennzeichnet er das unvollständige Basiselement mit drei Punkten ("...").
- Wenn sich der Modellierer hinsichtlich der Richtigkeit einer Spezifikation unsicher ist, so gibt er diesem Zweifel durch Annotation eines Fragezeichens Ausdruck ("?"). Von dieser Möglichkeit kann z.B. Gebrauch gemacht werden, wenn Unklarheit über die Angemessenheit einer Benennung eines Elements besteht oder über die gewählte Zerlegung in Subelemente besteht.

- Wenn sich der Modellierer nicht einmal sicher ist, ob seine Spezifikation unvollständig ist oder nicht, kann er diesen Fall durch drei Fragezeichen ("???) gekennzeichnet.

Folgende Abbildung gibt ein Beispiel für vages Modellieren:

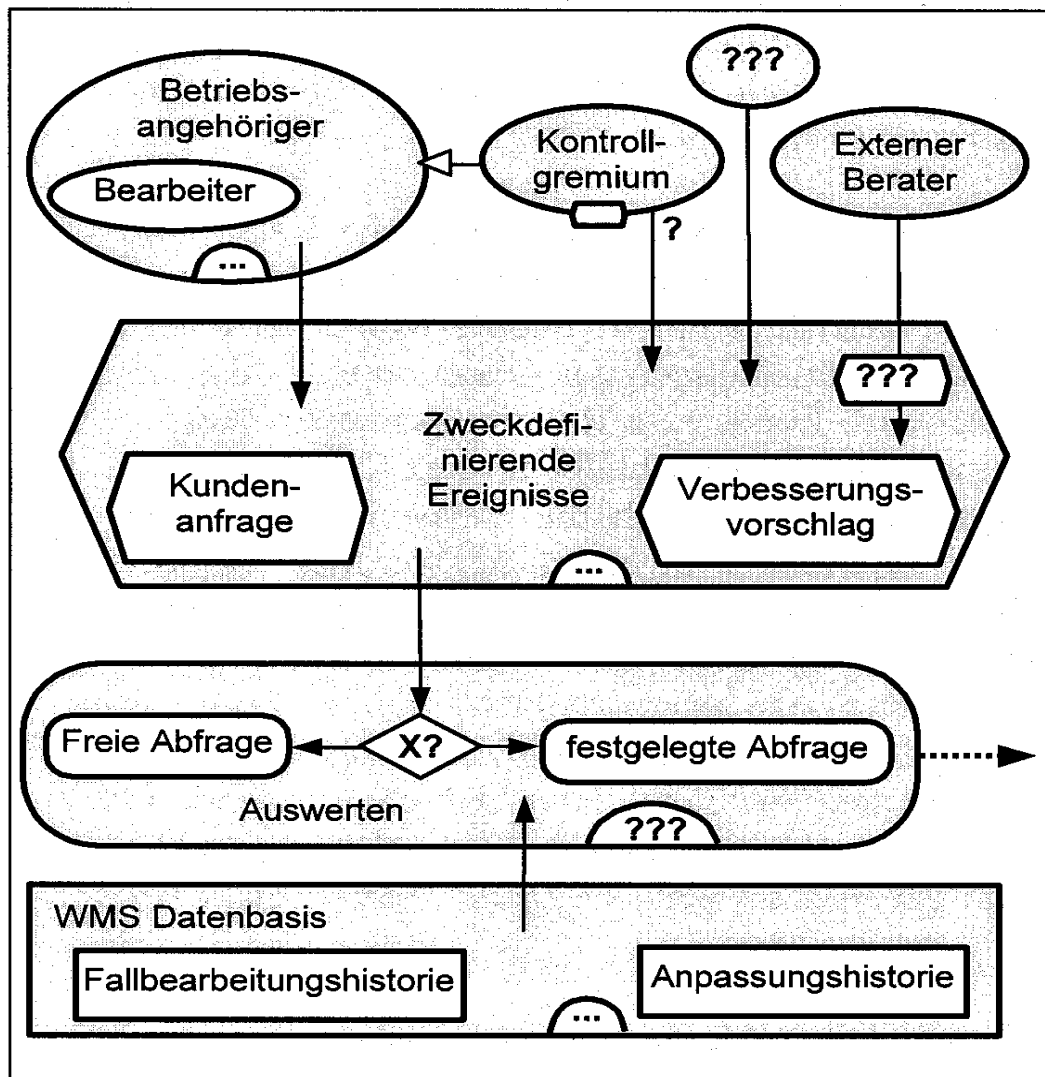


Abbildung 7: Beispiel für vages Modellieren [HHL99]

In dieser Abbildung werden die Sub-Elemente von Betriebsangehöriger und von zweckdefinierenden Ereignissen unvollständig spezifiziert. Beim Auswerten wird durch ("???) zum Ausdruck gebracht, daß eine Unsicherheit hinsichtlich Sub-Aktivitäten besteht. D.h. der Modellierer ist sich nicht sicher, ob noch zusätzliche Sub-Aktivitäten anzugeben sind. Außerdem wird die Richtigkeit des Kontektors XOR bezweifelt, da beide Formen der Abfrage kombiniert werden können. Der

Modellierer möchte den Modifikator, der die Bedingung der Existenz eines Kontrollgremiums angibt, nicht näher spezifizieren. Weiterhin ist er sich, bezüglich der Auswertungsmöglichkeit durch externe Berater unsicher. Die Unsicherheit besteht darin, ob sich hier überhaupt eine Bedingung angeben läßt, womit die Angemessenheit der eingetragenen Relation insgesamt in Frage gestellt wird.

Einsatzkontext

Die Entwickler der Modellierungsmethode SeeMe formulieren, die Einsatzgebiete der Modellierungsmethode SeeMe wie folgt:

"At the moment, SeeMe is still an experimental method. However, it has found surprising interest by certain consulting companies as well as by scientist who have the problem to model socitechnical systems in fields such as knowledge management or internet services for public administrations." [Herr99a]

Die Modellierungsmethode SeeMe kann zur Modellierung von Geschäftsprozessen, zur mitarbeiterorientierten Anforderungsanalyse, zur Unternehmensmodellierung herangezogen werden. Die in SeeMe-Methode enthaltenen Konzepte können auch zur Erweiterung anderer Modellierungsmethoden verwendet werden. Die eigenen Erfahrungen in einem Projekt schildert der Entwickler der Methode Herrmann, wie folgt:

"In a project with a training organisation we used SeeMe-Diagramms to suggest usage scenarios that integrated the design of a Know-How-Repisotory and the development of central business processes in the organization. Furthermore, we used SeeMe to capture and to compare results from some ethnographic studies of working processes." [Her99a]

SeeMe kann bei der sozial verträglichen Einführung von Informations- und Kommunikationssystemen sowie bei der ergonomischen Bewertung von Software eingesetzt werden. Sie kann auch bei der Modellierung und Planung von Geschäftsprozessen und bei der Einführung von Workflow-Management-Systemen Hilfestellung bieten. Bei der Modellierung mit SeeMe ist es auch möglich, die mitarbeiterorientierten Aspekte bzw. arbeitswissenschaftliche Kriterien wie KABA-Verfahren bei der Modellierung einzubeziehen.

3.2.3. ARIS

Das Konzept ARIS (Architektur integrierter Informationssysteme) und das Analyse- und Modellierungswerkzeug ARIS-Toolset bauen auf Ergebnissen des von Prof. Scheer gegründeten Forschungsinstituts für Wirtschaftsinformatik (IWi) an der Universität des Saarlandes auf. Das System ist von der IDS Scheer AG, die 1985 von Prof. Scheer gegründet worden ist, im Jahre 1994 für den internationalen Vertrieb freigegeben worden [Sch96]. Seit dem wird es an vielen Universitäten und Firmen in Europa, USA, Südafrika, Brasilien und Asien zur Unterstützung von Forschung und Lehre auf dem Gebiet der Unternehmensorganisation und der betriebswirtschaftlichen Informationsverarbeitung eingesetzt (15000 verkaufte Lizenzen, Stand Oktober 1999). ARIS wird zur Zeit in vielen großen Unternehmen (Mercedes Benz, Deutsche Telekom, Bayerischer Vereinsbank...) eingesetzt. Die IDS Scheer AG beschäftigt 845 Mitarbeiter (Stand Oktober 1999) [IDS99].

ARIS kennzeichnet sich durch ihre Orientierung an die Dokumentation von Standardsoftware. In diesem Zusammenhang ist insbesondere die Zusammenarbeit der IDS-Scheer AG mit den erfolgreichen Unternehmen SAP-AG zu nennen. Die Zusammenarbeit im wesentlichen beinhaltet die gemeinsame Entwicklung von Standardsoftware-Komponenten. Darüber hinaus ist die IDS Scheer, die von SAP-AG empfohlene Partner für die Einführung von R/3-Systemen. Die ereignisgesteuerten Prozeßketten dienen in ARIS zur Modellierung von Geschäftsprozessen. Sie haben sich in der Praxis zumindest im deutschsprachigen Raum als Beschreibungsmittel für betriebliche Abläufe etabliert [Sch96], [IDS99].

Die IDS Scheer GmbH hat im Herbst 1997 das Modellierungswerkzeug ARIS EasyDesign, ein Modellierungswerkzeug für die Fachabteilungen auf dem Markt gebracht. Sie soll insbesondere den Einstieg in Lernorganisation mit Leistungseinheiten unterstützen. Die IDS-Scheer AG ist seit 1999 ist an der Börse notiert.

Entstehungsgeschichte

Ausgangspunkt für die Entwicklung des ARIS-Konzeptes waren insbesondere die Erfahrungen von Prof. Scheer im Bereich der organisatorischen Einführung von Standardsoftware. Durch seine Erfahrungen stellte er fest, daß bei einem betriebswirtschaftlichen Standardsoftwaresystem das Verhältnis der Kosten für die organisatorische Einführung zu den Anschaffungskosten der Software 5:1 beträgt. Es kam hierdurch zu einer Vision, dieses Kostenverhältnis durch die Entwicklung von Konzepten, Methoden und Tools zur Unterstützung des Life Cycles der Software-einführung drastisch zu senken.

Um diese Vision realisieren zu können, wurde dann ein Rahmenkonzept (Architektur) zur vollständigen Beschreibung von Anwendungssoftwaresystemen entwickelt. In die Architektur werden die geeignet erscheinenden Methoden eingeordnet bzw. neue Methoden (VKD¹ und EPK²) entwickelt. Neben diesen konzeptuellen Arbeiten sah Prof. Scheer auch die Wichtigkeit der Sammlung von Erfahrungen mit der Eignung der Methoden. Diese Erfahrungen wurden insbesondere mit der SAP AG gesammelt.

Die ersten Erfahrungen mit der Beschreibung von Geschäftsprozessen machte Scheer, mit der von ihm entwickelten Methode Vorgangskettendiagrammen (VKD). Die Methode wurde 1981/82 in einem Kooperationsprojekt mit dem Unternehmen Villeroy & Boch erprobt. Sie floß später als Modellierungsmethode in das ARIS-Konzept ein.

Die Entstehung der Methode ereignisgesteuerter Prozeßketten geht auf die Zusammenarbeit von IDS und SAP-AG zurück. Im Jahre 1988/89 startete IDS-Scheer mit der SAP-AG ein Projekt, in denen die Systeme der SAP-AG durch Datenmodelle dokumentiert werden sollten. Sie stellten jedoch fest, „daß Datenmodelle für eine benutzerfreundliche Dokumentation gegenüber der Anwender noch nicht ausreichen, da der Benutzer in den Datenmodellen seinen Arbeitsablauf nicht anschaulich wiedergegeben findet.“ ([Sch96], S. 76) Aus diesem Hintergrund starteten sie ein zweites Projekt zur Modellierung von Geschäftsprozessen, die zur Entwicklung der EPK-Methode führte.

Theoretische Fundierung

Die EPK-Methode, die in ARIS zur Geschäftsprozeßmodellierung verwendet wird, basiert im wesentlichen auf die Petri-Netz-Theorie. Sie stellt eine Variante des Bedingungs-Ereignisnetzes, welche um logische Verknüpfungsoperatoren erweitert worden ist.

Es muß an dieser Stelle jedoch darauf hingewiesen werden, daß die EPK-Methode im Vergleich zu Petri-Netzen eine mangelnde syntaktischer und semantischer Fundierung besitzt. Sie hat weder eine explizit definierte Syntax noch eine explizite Semantik. Dies kann vielleicht auch damit begründet werden, daß bei der Entwicklung des ARIS-Konzeptes und ARIS-Toolsets die betriebswirtschaftlichen Problemstellungen und Theorie eine besondere Rolle gespielt haben. Prof.

¹ Vorgangskettendiagramme

² Ereignisgesteuerten Prozeßketten

Dr. Scheer bringt dies in einem Vergleich von ARIS mit der objektorientierten Modellierung wie folgt zum Ausdruck:

„Da die objektorientierte Modellierung mehr Gedanken zur Systementwicklung folgt, besitzt sie keine spezifisch betriebswirtschaftlichen Ausrichtung. Bei ARIS wurde aber von vornherein ein betriebswirtschaftlicher Geschäftsprozeß als Diskurswelt definiert. Auch die Aufnahme betriebswirtschaftlicher Gedanken der Produktionstheorie, Prozeßkostenrechnung und Unternehmensorganisation unterstreicht diesen Bezug.“ ([Sch98] S. 137)

In letzter Zeit sind Bemühungen zu beobachten, die EPKs mit objektorientierten Methoden zu verbinden bzw. zu kombinieren. In welche Richtung diese gehen sollte, bringt Scheer folgendermaßen zum Ausdruck:

„Auf keinen Fall wird aber ein strenger Gegensatz zwischen ARIS und der objektorientierten Modellierung angestrebt. Vielmehr sollten auch bei der Anwendung der objektorientierten Modellierung für eine konkrete Systementwicklung dem Benutzer die mehr betriebswirtschaftlich ausgerichteten ARIS-Modellsichten zum besseren und leichteren Verständnis geboten werden.“ ([Sch98a], S. 137)

Um die eingeschränkte Anwendbarkeit der EPKs zur formalen Analyse und zur Simulation aufzuheben, werden Anstrengungen unternommen die EPKs zu erweitern. Diese gehen davon aus, daß für Probleme, die im Rahmen des Einsatzes ereignisgesteuerter Prozeßketten auftreten, in der Petri-Netz Theorie bereits fundierte und evaluierte Lösungen vorliegen. Es wird dabei versucht, Regeln für die Übersetzung einer EPK in einem Petri-Netz zu formulieren. Anhand dieser Regeln soll dann die Übersetzung automatisch ablaufen. In diesem Zusammenhang wird ein Vergleich zwischen der Methode der ereignisgesteuerten Prozeßketten mit Petri-Netzen wie folgt lauten:

"Vergleicht man die Prozeßmodellierung mit dem Bau eines Hauses, so ist die EPK-Methode eine Benutzeroberfläche, damit der Bauherr dem Architekten seine Wünsche mitteilen kann. Petri-Netze haben dagegen die Funktion des Statikers, welcher die Konstruktion auf ihre Tragfähigkeit prüft." ([LSW97 S. 3])

Basiselemente und ihre Verknüpfung

Es wird als Modellierung in ARIS, „das Manipulieren von Elementen zur Beschreibung von Geschäftsprozessen in den von ARIS definierten Sichten und Phasen und mit den dort zugelassenen Begriffen und Methoden“ ([Sch98a], S. 119) verstanden.

Bevor die Basiselemente der Methode ereignisgesteuerter Prozeßketten erläutert werden, soll an dieser Stelle zunächst die Sichten des ARIS-Hauses kurz dargestellt werden. Dies scheint aus zwei Gründen notwendig. Erstens zeigt es, welcher Aspekte bei der Modellierung explizit betrachtet werden und zweitens dient es zur Einordnung der EPK-Methode im ARIS.

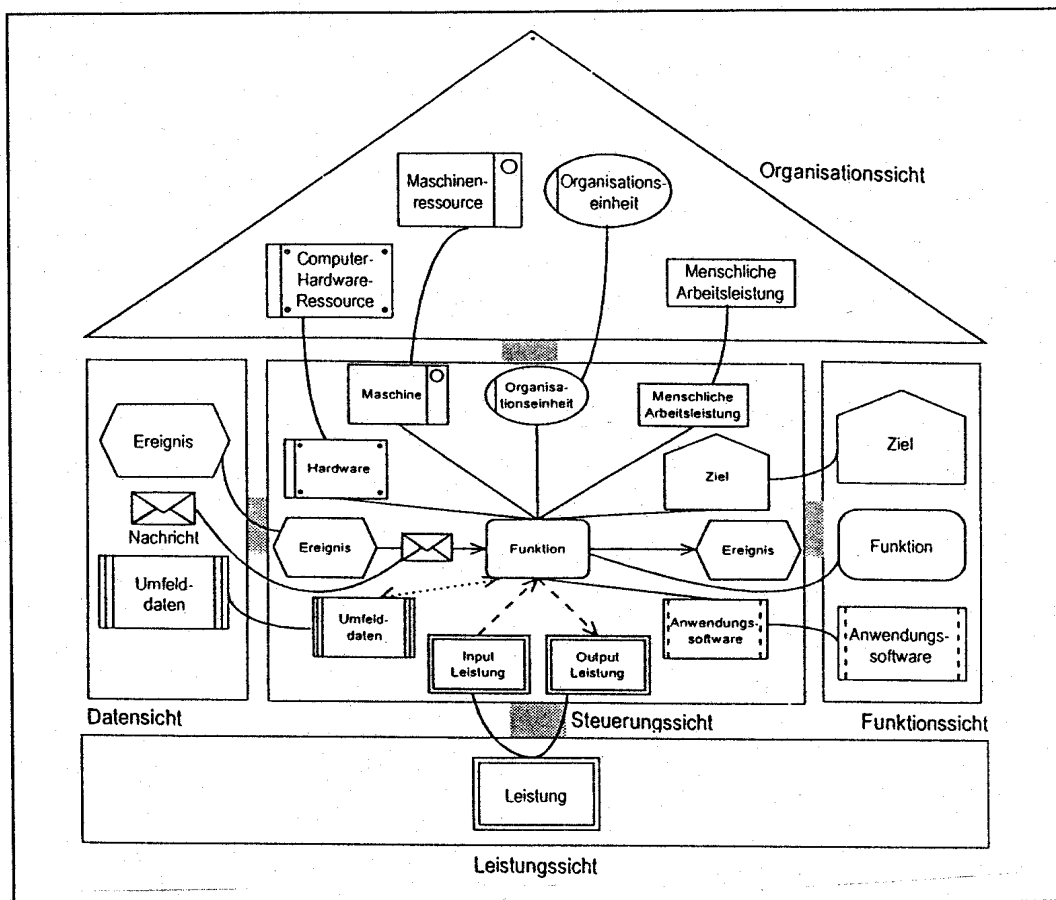


Abbildung 8: Sichten des ARIS-Hauses [Sch98a]

Die wesentliche Sicht in ARIS bei Unternehmensbeschreibungen ist die von Geschäftsprozessen. Diese Sicht wird wegen ihrer Komplexität in fünf Sichten unterteilt, die jeweils eigene Modellierungs- und Entwurfsmöglichkeiten darstellen. Im folgenden werden diese fünf Sichten kurz erläutert. Die Steuerungssicht und die ereignisgesteuerten Prozeßketten werden ausführlicher behandelt.

Modellierung der Funktionssicht: Im ARIS-Konzept werden die Funktionen als eigenständige Sicht auf einen Geschäftsprozeß behandelt. Eine Funktion wird dabei als eine Verrichtung an

einem Objekt zur Unterstützung einer oder mehrerer Ziele definiert. Wegen dieser engen Verbindung werden Ziele ebenfalls zur Funktionssicht zugeordnet. Auf dieser Sicht werden die Vorgänge, die Input-Leistungen zu Output-Leistungen transformieren, zusammengefaßt. Die Begriffe Funktion, Vorgang oder Tätigkeit werden synonym verwendet. Funktionen werden durch Rechtecke mit abgerundeten Ecken dargestellt. Funktionen können in ARIS auf unterschiedliche Verdichtungsstufen beschrieben werden. Oberste Verdichtungsstufe und damit Ausgangspunkt der Betrachtung sind komplexe Funktionsbündelungen. Um diese Komplexität zu reduzieren, wird ein Funktionsbündel in Teilfunktionen gegliedert. Bei der ARIS-Architektur wird eine strikte Trennung der Betrachtungsfelder verfolgt. Innerhalb der Funktionssicht werden nur solche Darstellungsmittel behandelt, die Verbindungen zwischen Funktionen aufzeigen. So werden z.B. Verbindungen zwischen Funktionen und Daten in der Steuerungssicht von ARIS dargestellt.

Modellierung der Organisationssicht: Unter Organisation wird in ARIS, Regelungen zur Koordinierung menschlicher und maschineller Aufgabenträger unter Beachtung der Unternehmensziele verstanden. Die Organisationssicht in ARIS beschreibt die Aufbauorganisation, also die Organisationseinheiten mit den zwischen ihnen bestehenden Kommunikations- und Weisungsbeziehungen. In dieser Sicht werden die Aufgabenträger, menschliche Arbeitsleistung, sowie die maschinelle Aufgabenträger, Betriebsmittel und Computer-Hardware zugeordnet. Die Organisationsstruktur wird in ARIS mit Hilfe von Organigrammen beschrieben.

Modellierung der Datensicht: Die Beschreibung der Datenobjekte, die von Funktionen manipuliert werden, erfolgt in der Datensicht. Es besteht bei den Datenobjekten, die von nachfolgenden Organisationseinheiten als Informationsdienstleistung anerkannt werden, eine Überschneidung zur Leistungssicht. In dieser Sicht können unterschiedliche Objekte mit unterschiedlicher Granularität verwendet werden. Die Datenobjekte, die in der Datensicht zusammengefaßt sind, beschreiben einerseits Ereignisse und Nachrichten, die den Ablauf des Geschäftsprozesses steuern (Kontrollfluß). Andererseits werden durch Datenobjekte, Umfeldzustände des Geschäftsprozesses dargestellt.

Modellierung der Leistungssicht: In ARIS werden Leistungen als Ergebnisse von Prozessen betrachtet. Der Bedarf nach Leistungen stellt auch die Veranlassung der Prozeßausführung. Die Beschreibung der Leistungen besitzt nach Scheer eine Kernfunktion innerhalb der Geschäftsprozeßbeschreibung. Der Leistungsbegriff umfaßt unterschiedliche Leistungsarten wie Sach- und Dienstleistungen. Die Leistungssicht wird durch die Modellierung von Produktionsfaktoren be-

schreiben. Zur Modellierung der Leistungssicht werden Produktbäume (bzw. Produktnetze) mit der logischen Kantenbeziehung „besteht aus“ verwendet. Wesentliche Merkmale von Leistungen sind, daß sie außerhalb der erzeugenden Stelle benötigt werden, also ein Bedarf für sie besteht und dafür ein Preis anerkannt wird. Dabei ist es unerheblich, ob diese Kunden-Lieferanten Verhältnis zu externen Partnern oder zwischen internen Organisationseinheiten besteht.

Modellierung der Beziehungen zwischen den Sichten (Steuerungssicht)

Die Steuerungssicht in ARIS hat die Aufgabe, die zunächst getrennt behandelten Sichten (Funktion, Organisation, Daten und Leistung) wieder zu verbinden. Die Dynamik des Systems wird durch die Ereignissteuerung und den Nachrichtenfluß ausgedrückt. Die Funktions-, Organisations-, Daten- und Leistungssicht beschreiben dagegen die statische Sicht auf das System. Die Steuerungssicht in ARIS betrachtet alle strukturellen Zusammenhänge der Sichten und zusätzlich die dynamischen Verhaltensaspekte des Geschäftsprozeßflusses.

Ereignisgesteuerte Prozeßketten

Mit der EPK - Methode wird im allgemeinen die zeitlich logische Abfolge von Funktionen im Sinne eines Geschäftsprozesses als Prozeßkette dargestellt. Dabei können für jede Funktion die Start- und Endereignisse angegeben werden. Ereignisse lösen Funktionen aus, und sind Ergebnisse von Funktionen.

Ein ***Ereignis*** wird durch das Eintreten eines betriebswirtschaftlich relevanten Zustandes eines Informationsobjektes definiert, der den weiteren Ablauf des Geschäftsprozesses steuert oder beeinflusst. Im Gegensatz zu einer Funktion, die ein zeitverbrauchendes Geschehen darstellt, ist ein Ereignis auf einen Zeitpunkt bezogen. Durch das Hintereinanderschalten dieses Ereignis-Funktionswechsels entstehen dann ereignisgesteuerte Prozeßketten.

Die logischen Abhängigkeiten können durch die Verwendung von ***Verknüpfungsoperatoren*** abgebildet werden. Es besteht weiterhin die Möglichkeit, daß die Eingänge und Ausgänge des Operators logisch verknüpft werden können. Dabei können sowohl die Ereignisse als auch Funktionen als Eingänge in einen Verknüpfungsoperator verwendet werden und somit den Start von Funktionen bzw. das Eintreten von Ereignissen steuern. Ereignisse werden in der Datensicht erfaßt. Insofern ist die Beschreibung der Ereignissteuerung von Prozessen eine Verbindung von

Daten- und Funktionssicht. Das ARIS- Toolset bietet die Möglichkeit zur Modellierung einer „erweiterten ereignisgesteuerten Prozeßkette“ (eEPK). Neben der Darstellung der Ereignisse und Funktionen können auch die organisatorischen Einheiten, die sie bearbeiten, sowie die verwendeten Daten im gleichen Modell dargestellt werden. Sie werden mit einem ungerichteten Pfeil mit den Funktionen verbunden.

Im folgenden werden die Elemente der Methode "erweiterten ereignisgesteuerten Prozeßketten" zusammengefaßt:

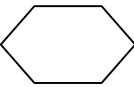
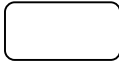



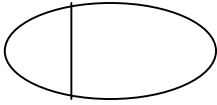

| Bezeichnung | Symbol | Definition |
|---|---|---|
| Ereignis |  | Ein Ereignis beschreibt das Eintreten eines Zustandes, der eine Folge bewirkt. |
| Funktion |  | Die Funktion beschreibt die Transformation von einem Eingangszustand in einem Zielzustand |
| Verknüpfungen |  | Die Verknüpfungsoperatoren beschreiben die logischen Verbindungen zwischen Ereignissen und Funktionen |
| Kontrollfluß |  | Der Kontrollfluß beschreibt die zeitlich logischen Abhängigkeiten von Ereignissen und Funktionen |
| Informationsobjekt |  | Das Informationsobjekt ist eine Abbildung eines Gegenstandes der realen Welt |
| Organisatorischer Einheit |  | Beschreibung der Gliederungsstruktur einer Organisation |
| Organisatorische Einheiten Zuordnung |  | Die Ressourcen-/Organisatorische Einheiten Zuordnung beschreibt, welche Einheit (Mitarbeiter) oder Ressourcen die Funktion bearbeiten |

Tabelle 1: Elemente der erweiterten ereignisgesteuerten Prozeßketten

Einsatzkontext

Die Methode der erweiterten ereignisgesteuerten Prozeßketten wird inzwischen in verschiedenen Projekten zur Unternehmensmodellierung eingesetzt. [IDS99]. ARIS kann insbesondere dort

eingesetzt werden, wo in der Regel komplexe Unternehmensprozesse dokumentiert und ausgewertet werden müssen. Im einzelnen kann sie,

- bei der Optimierung organisatorischer Veränderungen im Rahmen der Business Process Reengineering (BPR),
- zur Speicherung von Organisationswissen, z.B. in Form von Referenzmodellen
- bei der Erstellung der Prozeßdokumentation zur ISO-9000 ff. Zertifizierung,
- bei der Berechnung der Kosten von Geschäftsprozessen,
- bei der Einführung und Anpassung von Standardsoftware oder Workflow-Systemen,
- bei der Erstellung von Softwareferenzmodellen zum Abgleich von Anforderungen und Ist-Funktionalität,
- bei der Erstellung von Branchenreferenzmodellen zur Darstellung von typischen Abläufe einer Branche,
- bei der Erstellung von Vorgehensmodellen, z.B. zur Dokumentation des Softwareentwicklungsprozesses
- in Workflow-Systemen, die die eEPK Modelle ausführen

Aus der Sicht des Softwareentwicklers fehlt der EPK-Methode allerdings eine formale Grundlage. Um dieses Defizit auszugleichen, wurden Ansätze entwickelt, die die eEPKs auf Petri-Netze abbilden. Durch diese Abbildung wird einem eEPK-Modell eine eindeutige Semantik gegeben, die wiederum als Basis zur maschinellen Verifikation dienen kann [Sza98].

Mit Hilfe der EPK-Methode, kann der Zusammenhang zwischen Ablauf- und Aufbauorganisation untersucht werden. Es eignet sich auch zur Schwachstellenanalyse. Es kann z.B. durch die Zuordnung der organisatorischen Einheiten die Anzahl der zu überwindenden Schnittstellen in Geschäftsprozessen ermittelt werden.

ARIS eignet sich insbesondere für die Unternehmensberater, Organisationsgestalter Informationssystementwickler und für die Modellierung in Fachabteilungen. Sie ist jedoch nicht für Implementierungszwecke geeignet. Sie eignet sich für die Analyse der vorhandenen Abläufe, aber nicht für die Implementierung der Systeme.

3.2.4 Action/Workflow

Dieser Ansatz wurde von Terry Winograd und Fernando Flores entwickelt. Terry Winograd beschäftigte sich viele Jahre mit Informatik und künstliche Intelligenz am Massachusetts Institute of Technology (MIT), an der Stanford University und am XEROX Palo Alto Research Center. Fernando Flores hat auf höchster Regierungsebene Erfahrungen mit gesellschaftlicher und politischer Organisationen gemacht. Er war Wirtschafts- und Finanzminister zwischen 1970 und 1973 unter der Regierung Salvador Allende in Chile [WF89].

Im Jahre 1983 gründeten sie zusammen die Firma „ActionTechnologies“. Die Firma entwickelte zunächst MHS (Message Handling System), das verschiedene E-Mail Clients an ein Netzwerk anbinden konnte. Das System wurde mittlerweile an die Firma Novell verkauft. Später entwickelte Action Technologies das E-Mail System „Coordinator“, das aber wenig erfolgreich war, da zum einen E-Mail zu diesem Zeitpunkt noch nicht sehr weit verbreitet war und zum anderen das System den Benutzern zu strenge Regeln für die Kommunikation vorgab. Schließlich konzentrierten sie sich auf die Unterstützung von Geschäftsprozessen [AT99].

Entstehungsgeschichte

Die Erkenntnis, daß viele Ansätze zur Geschäftsprozeß- und Workflomodellierung sich hauptsächlich die Informationsverarbeitung in den Vordergrund rücken, veranlaßte Winograd/Flores zur Entwicklung eines neuen Ansatzes:

"Most current approaches to workflow management are structured around the domain of information processes. They begin with a class of information objects, such as forms or stored images, and define workflows as a sequence of actions to be done on those objects." ([FFM92], S. 282)

Nach der Meinung der Entwickler dieses Ansatzes, sollte es vielmehr bei der Analyse und der Modellierung, um die Erkennung von Zusammenhängen gehen. In diesem Sinne schreiben sie:

„When an analyst first asks people in an organisation „What is the work here?“, the natural response is to start looking at the forms and procedures. We explicitly reject this, ignoring the forms and asking „What are you actually doing“ ([FFM92], S.288)

Drei Punkte waren bei der Entwicklung dieses Ansatzes besonders maßgebend:

- Die Interaktion zwischen Menschen sollte in den Mittelpunkt gestellt werden.
- Der Prozeß sollte grundlegend von den Erfüllungsbedingungen abhängig sein, die seinen Ablauf lenken. Diese Erfüllungsbedingungen sollten die Basis für die Entscheidung bilden, ob der Prozeß erfolgreich abgeschlossen wurde oder nicht.
- Ein Geschäftsprozeß sollte nicht beendet betrachtet werden, wenn die Ausführung mit irgendeinem Ergebnis abgeschlossen wurde, sondern erst wenn das Ergebnis zufriedenstellend ist.

Theoretische Fundierung

Die theoretische Basis des ActionWorkflow-Ansatzes wird in dem Buch „Understanding Computers and Cognition. A new Foundation for Design“, das von Winograd/Flores geschrieben wurde dokumentiert [WF86]. Insbesondere die linguistische Sprechakttheorie, die sich vor allem mit der pragmatischen Dimension von Sprache beschäftigt, diente für sie zum Verständnis für die menschliche Handlung und menschliche Arbeit (language/action perspective).

„Our principal theoretical claim is that human beings are fundamentally linguistic beings. Action happens in language in a world constituted through language. What is special about human beings is that they produce, in language, common distinctions for taking action together. Language then is not a system for representing the world or for conveying thoughts and information. Language is ontology: a set of distinctions that allows us to live and act together in a common world.“ ([FGH88], S. 156)

Die linguistische Sprechakttheorie wurde von dem Philosophen J.L. Austin entwickelt und später von seinem Schüler Searle weiterentwickelt worden, in der die soziale Dimension von Sprache systematisch untersucht wird. Die Grundidee dieser Theorie kann vereinfacht mit dem Satz „Sprechen heißt handeln“ ausgedrückt werden. Die Handlungen werden dabei durch Sprechakte hervorgerufen. Unter einem Sprechakt werden von einem Sprecher gleichzeitig vollzogenen Äußerungen verstanden. Die Sprechakte rufen Handlungen hervor. Winograd/Flores übernehmen diese Gedanken und entwickelten ein Schema das sog. „Conversation for Action“. Sie betrachten die Reihenfolge von sog. Illokutionsakten im Laufe von Konversationen. Unter Konversation wird dabei, der Austausch von sprachlichen Äußerungen zwischen den handelnden Menschen, verstanden.

Die grundlegenden Punkte der Theorie des ActionWorkflow-Ansatzes kann wie folgt zusammengefaßt werden [WF89]:

- Organisationen existieren als Netze aus Direktiven (Aufforderungen, Befehle etc., mit denen der Zuhörer zu sprachlichen oder nichtsprachlichen Handlungen veranlaßt werden soll) und Kommissiven (Versprechen etc. mit denen sich der Sprecher auf bestimmte zukünftige Handlungen verpflichtet). Dabei beinhalten "Direktive", Anforderungen, Anfragen, Rücksprachen und Angebote; Versprechen, Zustimmung und Zurückweisen sind Formen der Kommissive.
- Zwangsläufig wird es zu Zusammenbrüchen kommen, und eine Organisation muß sich darauf einstellen. Die Bewältigung von Zusammenbrüchen erzeugt weitere Netze aus Direktiven und Kommissiven.
- Menschen innerhalb einer Organisation (einschließlich der Manager, aber nicht sie allein) geben mündliche oder schriftliche Äußerungen von sich, die eine Konversation entstehen lassen, wie sie das organisatorische Netzwerk erfordert. Sie beteiligen sich an der Erzeugung und Aufrechterhaltung des Kommunikationsprozesses. Den Kern des Prozesses bildet das Zusammenspiel sprachlicher Handlungen, die die unterschiedlichsten Formen von Verpflichtungen zur Folge haben.

Modellierung

Das Grundelement zur Modellierung ist in diesem Ansatz die ActionWorkflow Loop, aus der Arbeitsprozesse zusammengesetzt werden. Eine ActionWorkflow Loop umfaßt alle Interaktionsbeziehungen zwischen einem Kunden (engl. Customer) und einem Dienstleister (engl. Performer), welche im Zusammenhang mit dem Austausch von Leistungen entstehen. Die Begriffe Kunde und Dienstleister treffen auf die Menschen zu, die sowohl innerhalb einer Organisation als auch über die Organisationsgrenzen hinweg arbeiten.

Der ActionWorkflow Loop setzt sich aus vier Phasen zusammen:

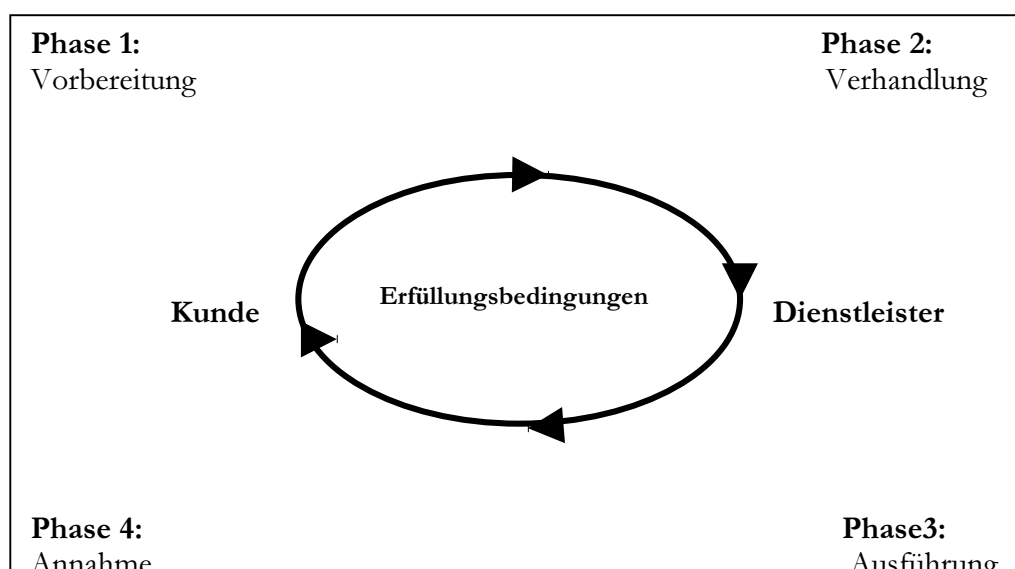


Abbildung 9: Action/Workflow-Loop

Vorbereitung (Preparation): Der Kunde fragt beim Dienstleister eine bestimmte Leistung nach, oder der Dienstleister bietet dem Kunden eine bestimmte Leistung zu bestimmten Bedingungen an.



Verhandlung (Negotiationen): Der Kunde und der Dienstleister einigen sich über die vom Dienstleister zu erbringende Leistung (Bedingungen).

Ausführung (Performance): Der Dienstleister erbringt eine Leistung und informiert darüber den Kunden

Annahme (Acceptance): Der Kunde überprüft die Leistung und informiert den Dienstleister über das Resultat der Überprüfung.

In jeder Phase sind weitere Aktionen möglich, wie etwa zusätzliche Verhandlungen, notwendige Klärungen und die Absprachen über die Veränderungen der Bedingungen bei den Beteiligten. Die Strukturen dieser Loop werden durch die Sprechakte definiert, durch den die Menschen sich koordinieren. Dabei spielen die Aktionen, die von Individuen zur Erfüllung von Bedingungen verrichtet werden, eine untergeordnete Rolle. Hiermit wird bei der Modellierung der Blickpunkt, anstatt auf die Aufgabenstrukturen auf die Koordinationsstrukturen gelenkt.

Da eine einzelne ActionWorkflow Loop kein besonders geeignetes Modell ist, um die gesamte Abläufe darzustellen, kann ein primärer Prozeß in jeder seiner Phasen, sekundäre Prozesse anstoßen. Durch Links (Pfeile) können Zusammenhänge zwischen unterschiedlichen Schleifen dargestellt werden. Workflows der ersten Zerlegungsebene werden als "Primary Workflows" bezeichnet. Die Workflows aller weiterer Zerlegungsebenen werden als "Secondary Workflows" genannt. Die Menge aller Secondary Workflows zusammen mit ihrer Primary Workflows wird als "Business Process" bezeichnet. Ein Workflow Schema im ActionWorkflow-Ansatz wird als "Business Process Map" bezeichnet.

Als Verknüpfungsoperatoren werden zwei Symbole verwendet. Hängt ein Teilprozeß von den Ergebnissen vorherigen Teilprozeß ab, so kann es mit dem Symbol  gekennzeichnet werden. Löst ein Teilprozeß wiederum mehrere weitere Teilprozesse, kann das Symbol  verwendet werden.

Die Erfüllungsbedingungen (Conditions of Satisfaction) beschreiben die Anforderungen, welche der Kunde an die Leistungen stellt, die vom Dienstleister erbracht werden sollen. Die Erfüllungsbedingungen (conditions of Satisfaction) können sowohl die Bedingungen (z.B. der Auftragsbestätigung muß innerhalb vier Tage erfolgen) als auch die Ziele, die mit der Handlung verbunden sind beinhalten.

Einsatzkontext

ActionWorkflow-Ansatz kann zur Analyse, Modellierung und Optimierung bzw. Reorganisation von kooperativen Arbeitsprozessen herangezogen werden. Sie eignet sich insbesondere dort, wo der Interaktion zwischen Menschen im Vordergrund steht, und der Prozeß schwachstrukturiert ist. Sie kann von den Unternehmensberatern, Organisationsentwicklern, Systemanalytikern zum Zwecke der Analyse und Reorganisation von Geschäftsprozessen eingesetzt werden.

Weiterhin kann sie von SoftwareentwicklerInnen zur Groupwareentwicklung herangezogen werden. Die erstellten Modelle können mit Hilfe des Werkzeugs „Action/Application Builder zur Entwicklung von Anwendungen herangezogen werden. Der Application Builder gibt es in zwei Versionen - für Lotus Notes und SQL-Server. In der Lotus Notes Version erfolgt mit dem Application Builder nicht nur eine automatische Generierung der Prozeßdefinitionen sondern auch die Anwendungsentwicklung - mit automatischer Generierung von Formularen zur Daten-ein-/ausgabe und Sichten zum Anzeigen bestimmter Datenbestände. In der SQL Version werden neben der automatischen Generierung von Prozeßdefinitionen, die Anwendungsdaten und Formulare festgelegt. Der Application Builder stellt in der SQL Version auch eine Anwendungsprogrammierschnittstelle zur Verfügung, um Anwendungen an das Action/Workflow System anzubinden. Diese API stellt einen Satz an Funktionen zur Verfügung, die mit Hilfe geeigneter Software (Visual Basic, Visual C++, etc.) die Programmierung von Formularen und Sichten ermöglicht [AT99], [Mor92], [MWR92].

3.3 Ergebnisse des Vergleichs

In diesem Kapitel ist ein analytischer Vergleich der Ansätzen zur Kooperationsmodellierung durchgeführt. Die Ergebnisse dieses Vergleichs werden im folgenden tabellarisch dargestellt. Zunächst werden die Ansätze mit ihrer Namen, Kurznamen und Literaturhinweisen zusammengefaßt. Abschließend werden - entsprechend des Analyserasters – die Entstehungsgeschichte/Einsatzkontext und theoretische Fundierung der Ansätze sowie die Basiselemente mit ihren Verknüpfungselementen gegenübergestellt.

(1) Name, Kurzname und Literaturhinweis

| Name des Ansatzes | Kurzname | Literaturhinweise |
|---|-----------|--|
| Rollen-/Funktions-/Aktions-Netze | RFA-Netze | Horst Oberquelle: „Sprachkonzepte für benutzergerechte Systeme“ Springer, Berlin (1987) |
| Sozio-orientierte semi-strukturierte Modellierungsmethode | SeeMe | Thomas Herrmann et al.: „Modellieren mit SeeMe – Alternativen zur Trockenlegung feuchter Informationslandschaften“. In J. Desel et al. (Hrsg.) „Modellierung 99“, S. 59 - 74 |
| Architektur Informationssysteme | ARIS | August Wilhelm Scheer: „ARIS- Vom Geschäftsprozeß zum Anwendungssystem“ Springer, Berlin (1998) |
| Action/Workflow | - | R. Medina-Mora, T. Winograd, R. Flores, F.Flores: „The Action/Worklflow Approach to Workflow Management Technology“. In: Conference on CSCW'92, ACM, New York (1992), S. 281 - 288 |

Tabelle 2 : Namen, Kurznamen der Ansätze mit Literaturhinweisen



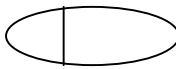
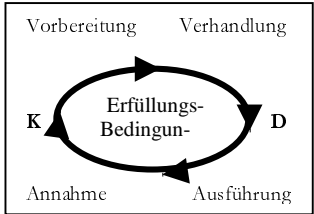

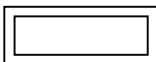
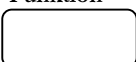
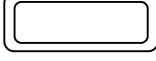
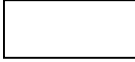


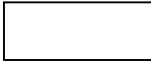
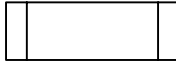



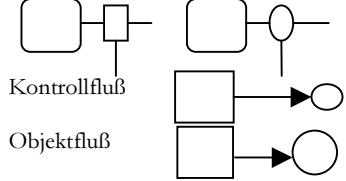
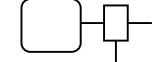


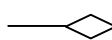
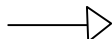











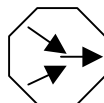
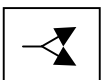
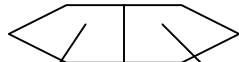
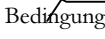
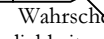
(2) Entstehungsgeschichte/Einsatzkontext und theoretischer Fundierung

| | RFA-Netze | SeeMe | ARIS | Action/Workflow |
|--------------------------------------|--------------|---|--|--|
| Entstehungsgeschichte/Einsatzkontext | Wissenschaft | sozialorientierte Forschungs- und Pilotprojekte | 1. Geschäftsprozeßmodellierung 2. Einführung von Standardsoftware | 1. Kooperationsunterstützung 2.Groupwareentwicklung |

| | | | | |
|-------------------------|--------------|--|------------------------|------------------|
| Theoretische Fundierung | Petri -Netze | "feuchte Informationslandschaften" nach „Goguen“ | erweiterte Petri-Netze | Sprechakttheorie |
|-------------------------|--------------|--|------------------------|------------------|

Tabelle 3: Einsatzkontext/Entstehungsgeschichte und theoretischer Fundierung der Ansätze

(3) Basiselemente und Ihre Verknüpfung

| RFA-Netze | SeeMe | ARIS | Action/Workflow |
|---|---|--|---|
| Rolle:  | Rolle:  | Organisatorische Einheit  | Action/Workflow-Loop  |
| Aktivität:  Komplexe Aktivität  Funktion   | Aktivität:  | Funktion:  | |
| Objekt:  | Entität  | Informationsobjekt  | |
| Zustand  | Inkrafttreten einer Relation  | Ereignis  | |
| Kooperations- und Interaktionsbeziehungen  Kontrollfluß  Objektfluß  | Relationen Standard  Aggregation  Vererbung:  Meta-Relation  | Kontrollfluß  Informations-Materialfluß  Organisatorische Einheiten Zuordnung  | Link  |
| | Konnektoren UND  ODER  entweder oder  | Verknüpfungsoperatoren UND  ODER  entweder oder  | UND  ODER  |
| | Modifikatoren  Bedingung  Wahrscheinlichkeit  | | |


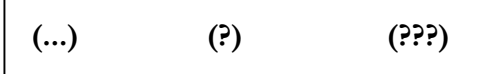
| | | |
|----------------------------|---|--|
| Erweiterungen in SeeMe: | Beabsichtigte Auslassung | Kennzeichnung unsicherer Information |
| |  |  |

Tabelle 4 : Gegenüberstellung der Basiselemente und ihre Verknüpfungen

Kapitel 4

Vergleich der Ansätze am Fallbeispiel "Prüfungsleistung Diplomarbeit"

In diesem Teil der Arbeit wird der praktische Vergleich der Ansätze durchgeführt. Zu diesem Zweck ist das Fallbeispiel „Prüfungsleistung Diplomarbeit“ ausgewählt und mit jedem Ansatz graphisch modelliert. Ein wesentlicher Grund für die Auswahl dieses Fallbeispiels besteht darin, daß in ihm eine Vielzahl von Personen beteiligt sind und die Kooperation zwischen den Beteiligten auf unterschiedliche Art und Weise - sowohl formal als auch informell - erfolgt. Als Grundlage für den Vergleich dienen einerseits „der Diplomprüfungsordnung für den Studiengang Informatik“ und andererseits die eigenen Erfahrungen, die im folgenden als Szenario zusammengefaßt werden. Es sei an dieser Stelle ausdrücklich darauf hingewiesen, daß dieser Vergleich sich nur auf das ausgewählte Fallbeispiel bezieht und von daher auch als ein bedingter Vergleich aufgefaßt werden muß. Das Ziel ist hierbei vielmehr, die Zusammenspiel der Basiselemente und ihre Verknüpfungen anhand des Fallbeispiels zu verdeutlichen. Die Ansätze sollen am Ende dieses Kapitels, im Hinblick auf die Unterstützung bei der Erstellung komplexer Modelle verglichen werden.

4.1. Fallbeispiel: "Prüfungsleistung Diplomarbeit"

An der Erstellung einer Diplomarbeit als Teil der Gesamtprüfungsleistung sind eine Vielzahl von Personen (Studierende, Betreuer bzw. Gutachter, Prüfungsamt bzw. entsprechende SachbearbeiterInnen in den Fachbereichen) beteiligt, die auf unterschiedliche Art und Weise kooperieren. Während dabei die Interaktion zwischen Studierenden bzw. Betreuer einerseits und der Prüfungsverwaltung andererseits formal geregelt abläuft, ist die Kooperation innerhalb eines Betreuungsverhältnisses von eher informellem und individuellem Charakter.

Wie die Kooperation zwischen Studierenden bzw. Betreuer und Prüfungsverwaltung ablaufen soll, wird durch die Diplom-Prüfungsordnung für den Studiengang Informatik von 17. April 1985 vorgegeben. Diese formal-juristische Sicht auf die Diplomarbeitstellungsprozesses wird im folgenden wiedergegeben.

§23 der Diplom-Prüfungsordnung für den Studiengang Informatik von 17. April 1985

- (1) Die Diplomarbeit kann als Einzel- oder Gruppenarbeit angefertigt werden.
- (2) Die Diplomarbeit ist Bestandteil der wissenschaftlichen Ausbildung. In Ihr soll der Kandidat eine theoretische oder praxisbezogene Aufgabe aus seiner Fachrichtung nach bekannten Verfahren und wissenschaftlichen Gesichtspunkten selbständig bearbeiten.
- (3) Der Diplomarbeit geht die Anfertigung einer Studienarbeit voraus; näheres regelt die Studienordnung
- (4) Der Kandidat schlägt einen Betreuer dem Prüfungsausschuß vor; dem Vorschlag ist soweit möglich und vertretbar entsprechen. Das Thema der Diplomarbeit wird von Betreuer aus dem Gebiet der Informatik gestellt; der Kandidat kann Vorschläge für das Thema der Diplomarbeit unterbreiten. Auf Vorschlag des Kandidaten ernennt der Prüfungsausschuß einen Zweitbetreuer; in begründeten Fällen kann der Prüfungsausschuß von den Vorschlägen abweichen. Betreuer und Zweitbetreuer sind gleichzeitig Prüfer für die Diplomarbeit; sie müssen die entsprechende Prüfungsberechtigung nach §10 dieser Prüfungsordnung besitzen; mindestens einer von ihnen soll Professor am Fachbereich Informatik der Universität Hamburg sein. Die Diplomarbeit darf mit Zustimmung des Prüfungsausschusses in einer Einrichtung außerhalb der Hochschule ausgeführt werden, wenn dort ein von Fachbereichsrat Informatik zugelassener Zweitbetreuer zur Verfügung steht.
- (5) Das Thema der Diplomarbeit ist vom Betreuer dem Prüfungsausschuß mitzuteilen. Dabei ist eine vom Betreuer der Studienarbeit unterzeichnete Bescheinigung über deren Abschluß vorzulegen. Der Ausgabezeitpunkt aktenkundig zu machen.
- (6) Der Schwierigkeitsgrad der Diplomarbeit ist so zu wählen, daß die Bearbeitung in einer Frist von zwölf Monaten abgeschlossen werden kann; die Frist beginnt mit der Bestellung des Betreuers und des Zweitbetreuers, sowie der Mitteilung Themas der Diplomarbeit an den Prüfungsausschuß. Auf Antrag des Kandi-

daten und im Einvernehmen mit dem Betreuer kann der Prüfungsausschuß die Bearbeitungsfrist in Ausnahmefällen um maximal sechs Monate verlängern. Die abgeschlossene Diplomarbeit ist vom Kandidaten fristgemäß dem Prüfungsausschuß einzureichen. Der Abgabetermin ist aktenkundig zu machen. Wird keine Arbeit vorgelegt, so gilt die Arbeit nach Ablauf der Fristen als abgebrochen. Bei überschreiten der vorgesehenen Friste unter Berücksichtigung von Verlängerung kann der Kandidat unverzüglich die bis dahin angefertigte Arbeit zur Begutachtung vorlegen; diese Arbeit ist sodann zu beurteilen.

- (7) Findet ein Kandidat trotz Bemühens keinen Betreuer für seine Diplomarbeit, so sorgt der Prüfungsausschuß auf Antrag des Kandidaten für ein Thema und die erforderliche Betreuung.
- (8) Der Kandidat kann einmal von der Diplomarbeit zurücktreten und ein neues Thema beantragen; geschieht dies innerhalb von zwei Monaten nach Beginn der Diplomarbeit, so gilt die erneute Diplomarbeit nicht als Wiederholung. Tritt der Kandidat später zurück aus Gründen, die er nicht zu vertreten hat, dann gilt dies nicht als Wiederholung; die Feststellung hierüber trifft der Prüfungsausschuß.
- (9) Die Diplomarbeit ist mit einer Erklärung des Kandidaten zu versehen, daß er diese (gegebenenfalls welchen Anteil an der Diplomarbeit) selbständig durchgeführt hat, und daß er keine anderen als die angegebenen Quellen und Hilfsmittel benutzt hat.
- (10) Die Diplomarbeit soll in einem Seminarvortrag vorgestellt werden.

Für den nachfolgenden Vergleich der Ansätze wird im allgemeinen sowohl die Informationen aus dem vorherigen Text, als auch folgendes Szenario (ohne Verlängerung), die auf eigenen Erfahrungen basieren, als Grundlage für die Modellierung verwendet:

Der Student möchte eine Diplomarbeit am Arbeitsbereich Softwaretechnik schreiben. Zu diesem Zweck besucht er ein Projektseminar, das in diesem Arbeitsbereich angeboten wird. Im Rahmen des Projektseminars bieten der/die VeranstalterInnen Themen zur Bearbeitung für eine Diplomarbeit. Der Student wählt hiervon ein Thema. Der Student und Betreuer konkretisieren das ausgewählte Thema. Zu diesem Zweck erstellt der Student Dokumente, wie Expose, Gliederung und Zeitplan. Der Student muß nun einen Zweitbetreuer aussuchen und die Diplomarbeit beim Prüfungsausschuß anmelden. Im Rahmen der Betreuung der Diplomarbeit finden zu verschiedenen Zeiten zwischen Betreuer und Student Treffen statt, wo über die Versionen der Diplomarbeit diskutiert wird. Der Betreuer hat hierbei die Aufgabe, die verschiedenen Versionen der Arbeit zu lesen und gegebenenfalls zu korrigieren. Die Diplomarbeit muß fristgerecht (innerhalb von 12 Monaten) beim Prüfungsamt abgegeben werden. Das Prüfungsamt versendet dann die Diplomarbeit an den Erst und Zweitbetreuer, zum Zwecke der Begutachtung und Benotung. Nach dem die Diplomarbeit begutachtet und benotet ist, wird das schriftliche Gutachten an das Prüfungsamt versendet. Der Student wird dann vom Prüfungsamt über die Note benachrichtigt.

Die verschiedenen Akteure, Aufgaben und die Gegenstände, die in dem Fallbeispiel vorkommen, werden im folgenden in Form einer Tabelle zusammengefaßt. Wo eine Kooperation zwischen Beteiligten stattfindet, kann aus der Spalte „Akteure“ entnommen werden. Da das Ziel

dieses Kapitels einen Überblick über die graphischen Modellierungsmöglichkeiten der Ansätze zu geben ist, wird hierbei von Details und Ausnahmefällen abgesehen. Es soll daher nicht als vollständig betrachtet werden.

| Aufgaben | Akteure | Gegenstände |
|--|--------------------------------|------------------------------------|
| Thema anbieten | Arbeitsbereich Softwaretechnik | Seminarangebot im KVV ³ |
| Thema konkretisieren | Student und Erstbetreuer | Expose, Gliederung, Zeitplan |
| Zweitbetreuer finden | Student | Expose, Gliederung etc. |
| Anmeldung der Diplomarbeit | Student | Anmeldungsformular |
| Prüfen der Anmeldung | Prüfungsausschuß | ausgefülltes Anmeldeformular |
| Betreuung | Erstbetreuer, Zweitbetreuer | Versionen der Arbeit |
| Teilergebnisse vorstellen und absprechen | Betreuer und Student | Expose, Gliederung, etc. |
| Lesen bzw. korrigieren der Teilergebnisse | Erstbetreuer und Zweitbetreuer | Expose, Gliederung, etc. |
| Anfertigung der Arbeit | Student | Versionen der Arbeit etc. |
| Abgabe der Diplomarbeit | Student | Diplomarbeit |
| Versendung der Diplomarbeit an die Betreuer | Prüfungsamt | Diplomarbeit Diplomarbeit |
| Begutachtung | Erstbetreuer und Zweitbetreuer | Diplomarbeit |
| Benotung | Erstbetreuer und Zweitbetreuer | Diplomarbeit |
| Mitteilung der Note | Prüfungsamt | Gutachten |

Tabelle 5: Fallbeispiel "Prüfungsleistung Diplomarbeit"

4.2 Modellierung mit den Ansätzen

4.2.1 Modellierung mit RFA-Netzen

Wie schon im Kapitel 5 beschrieben wurde, können mit RFA-Netzen sowohl die statischen als auch die dynamischen Aspekte modelliert werden. Bei der Modellierung mit RFA-Netzen müssen zunächst die Rollen, Funktionen, Handlungen/Aktionen und Objekte identifiziert werden.

³ Kommentiertes Veranstaltungsverzeichnis

Mit RFA-Netzen können die Beziehungen zwischen den Beteiligten im Modell explizit gemacht werden. Die folgende Abbildung stellt in gröbster Form die Kooperationsbeziehungen zwischen den Rollen wieder.

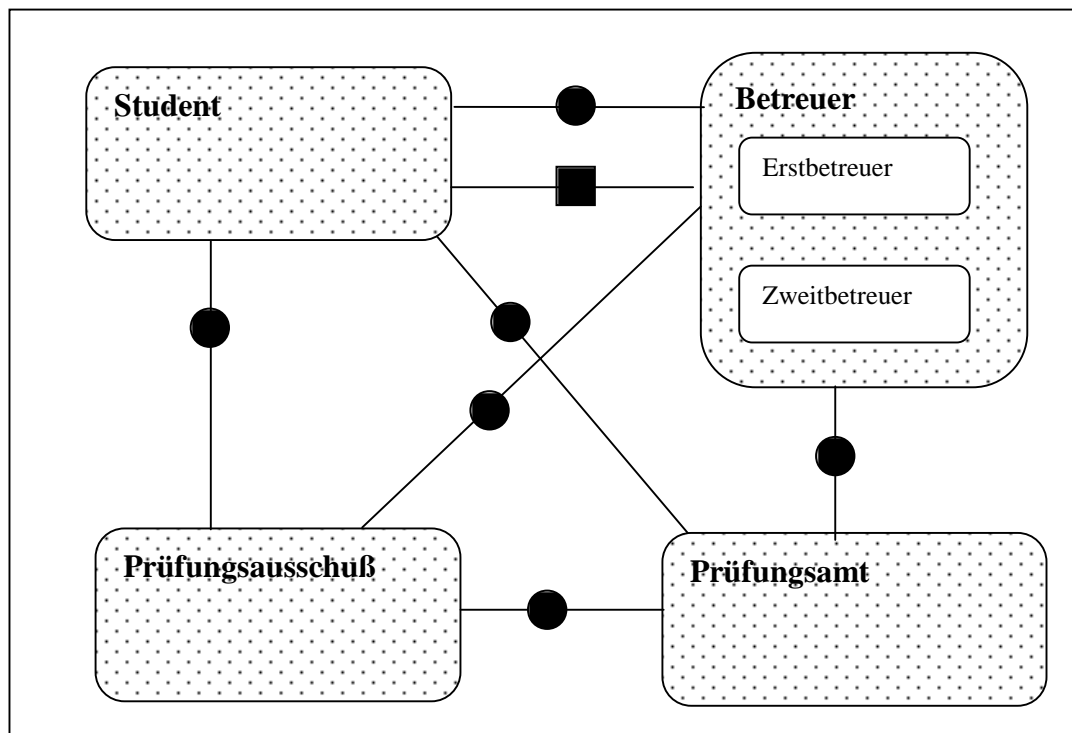


Abbildung 10: Kooperation zwischen Rollen (R-Netz)

In der Abbildung werden die Beteiligten - Student, Betreuer, Prüfungsverwaltung und ihre Kooperationsbeziehungen, mit Hilfe des Rollennetzes dargestellt. Sie können graphisch, in Form von Kreisen und Quadraten zum Ausdruck gebracht werden. So z.B. bedeutet das Quadrat zwischen Student und dem Betreuer, daß sie durch gemeinsame Handlungen kooperieren. Der Kreis hingegen soll darauf hinweisen, daß die Kooperation über Schnittstellen erfolgt. Somit kann aus der Abbildung hergeleitet werden, daß der Student und Betreuer sowohl direkt durch gemeinsame Handlungen (z.B. gemeinsamer Informationsaustausch im Gespräch, ...) als auch indirekt über Schnittstellen (z.B. Ablage der Dokumente im Fach des Betreuers im Sekretariat) kooperieren. Genauso ist die Kooperation zwischen Student und Prüfungsausschuß einerseits und Prüfungsamt andererseits zu interpretieren.

Der dynamische Ablauf - Zusammenspiel von Funktionsträgern, Handlungen und Objekten in Zeit und Raum - des Prozesses erfolgt in diesem Ansatz auf der Aktionsebene, mit Hilfe von Aktionsnetzen. Wie bereits erwähnt, kann mit RFA-Netzen sowohl der Kontrollaspekt (Kontrolle von Handlungen durch Funktionsträger) als auch der Objektaspekt (die Bearbeitung von Objekten in Handlungen) getrennt betrachtet werden. Im folgenden werden daher diese beiden Aspekte zunächst getrennt modelliert. Anschließend wird das erstellte Kontroll- und Objektnetz durch Überlagerung in einem Aktionsnetz überführt. Im folgenden werden aus Übersichtlichkeitsgründen das Kontrollnetz nur für die Rollen "Student" und "Betreuer" erstellt. Es besteht auch die Möglichkeit weitere Rollen (wie z.B. Zweitbetreuer, Prüfungsamt etc.) in das Modell aufzunehmen.

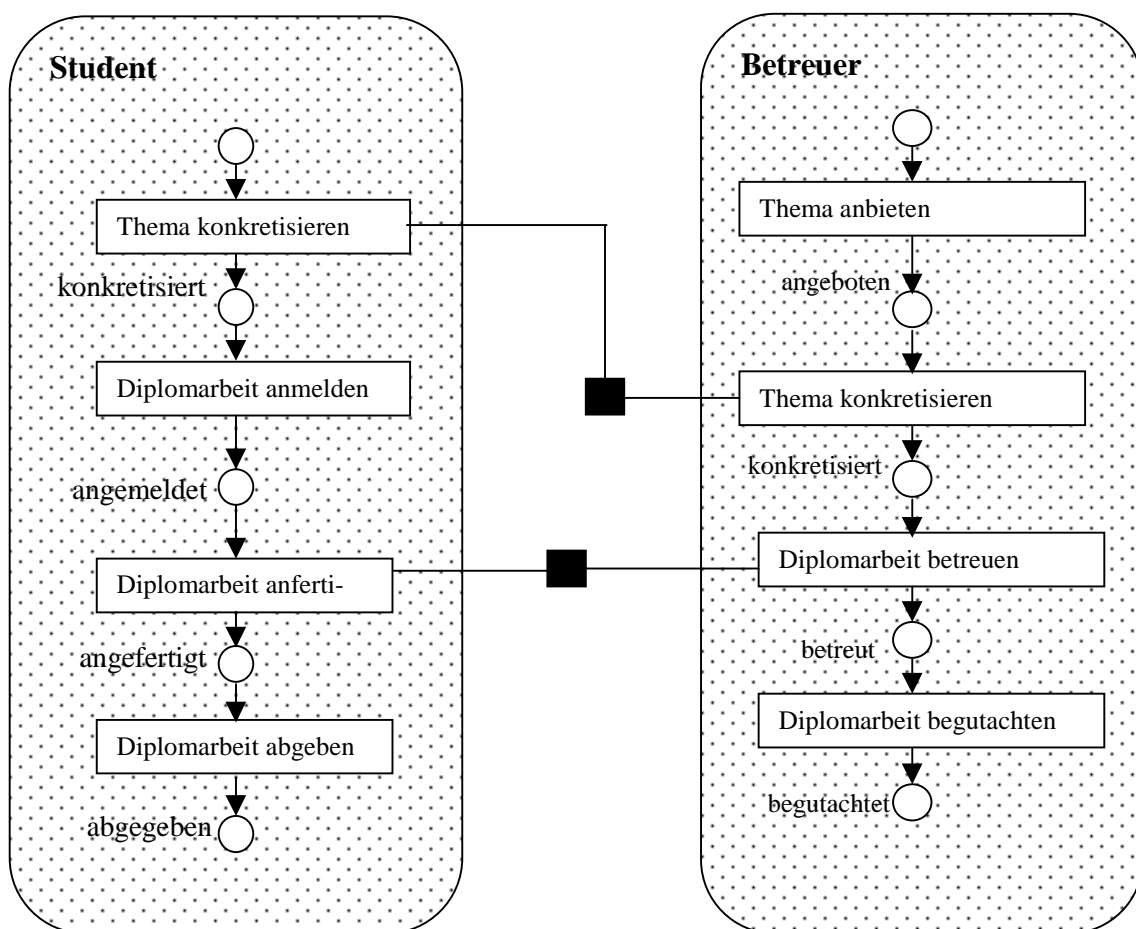


Abbildung 11: Kontrollnetz mit Rollenordnung (RA-Netz)

Die kleinen Kreise modellieren die (zeitlichen) Zustände, die Rechtecke hingegen die Tätigkeiten der Rollen. Die Tätigkeiten der Rollen "Student" und "Betreuer" werden hierbei elementar angenommen. Es besteht jedoch auch die Möglichkeit sie als komplex darzustellen (durch doppelte Umrandung). Die kleinen Quadrate sollen auf den gemeinsamen Handlungen der Rollen (z.B. Thema konkretisieren) hindeuten.

Der Objektfluß - die Bearbeitung von Objekten durch Handlungen - kann durch die Benutzung von Objektnetzen erfaßt werden.

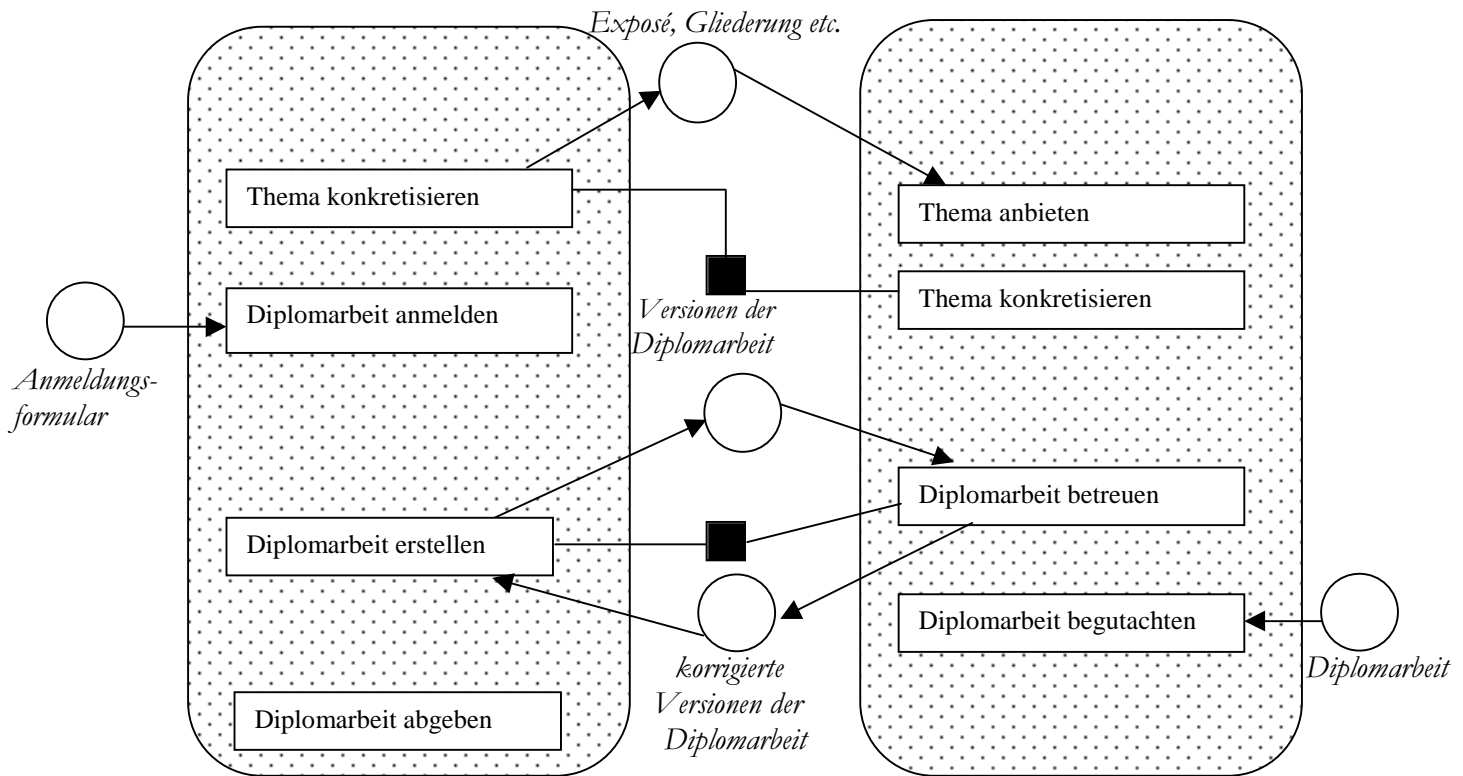


Abb. 12: Objektnetz mit Rollenzuordnung (RA-Netz)

Für die Erfassung des Objektaspektes können in diesen Netzen sowohl informale als auch formalisierte Darstellungen verwendet werden (In [Obe87] wird sehr ausführlich auf diese Möglichkeiten der Beschreibung von Objekten und von Operationen an Objekten eingegangen).

Um das Verhalten von Rollen bzw. Funktionen vollständig zu erfassen, können in diesem Ansatz Kontrollaspekt und Objektbearbeitung gemeinsam betrachtet werden. Dies führt dann zu den Aktionsnetzen. In diesen Netzen wird wegen der unterschiedlichen Darstellung von Kontrollfluß und Objektfluß, die Aufspaltung eindeutig definiert. Sowohl in Kontrollnetzen als auch in Objektnetzen besteht die Möglichkeit der präzisen Beschreibung der Dynamik durch Markierungen und Schaltregeln für Handlungen. Diese lassen sich auch ohne Probleme auf die Aktionsnetze

übertragen. Die zu dem Prozeß "Diplomarbeit" zugehöriges Aktionsnetz würde folgendermaßen aussehen:

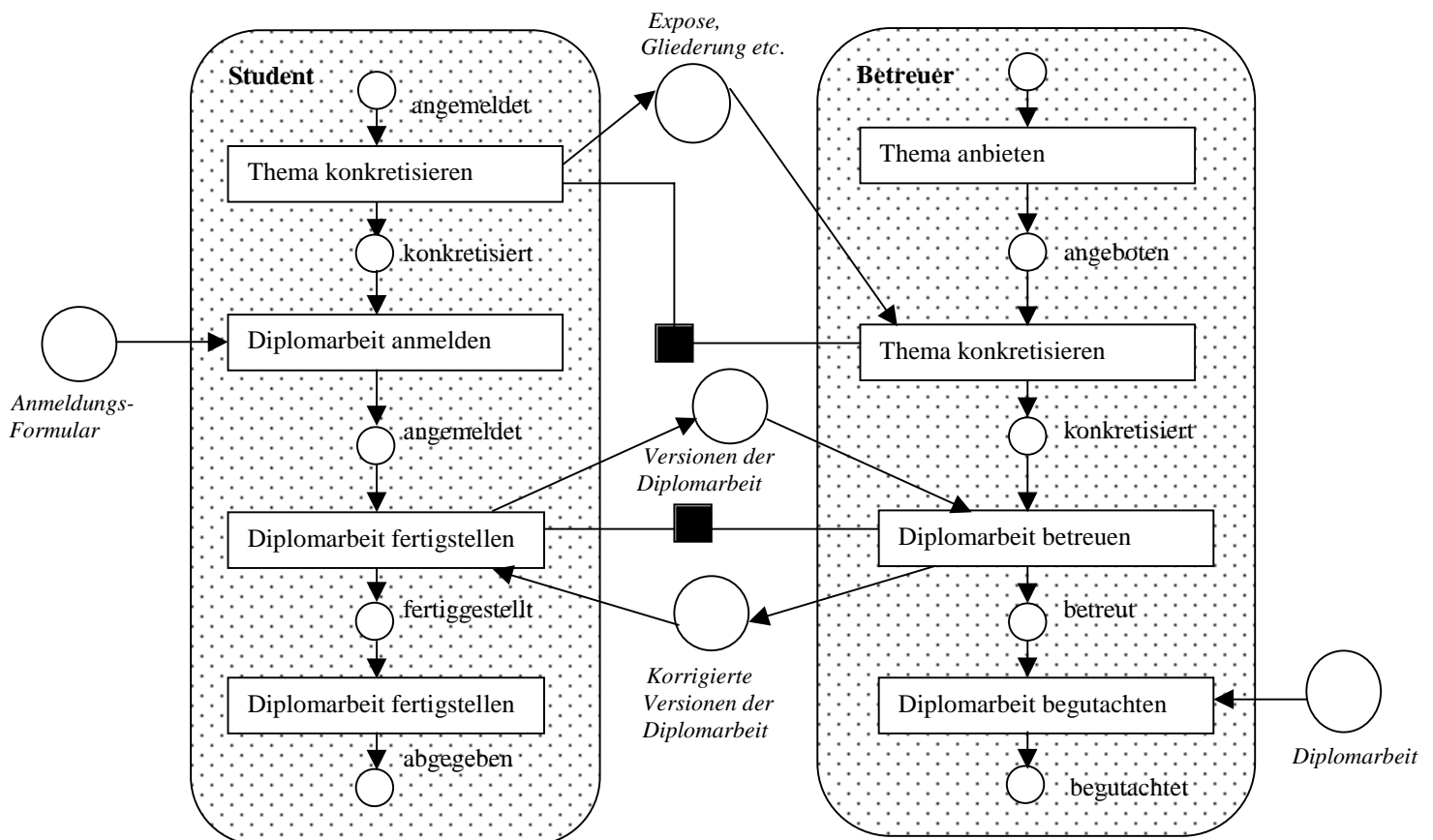


Abbildung 12: Aktionsnetz als Überlagerung von Objektnetz und Kontrollnetz

4.2.2 Modellierung mit SeeMe

Die Modellierungsmethode SeeMe erlaubt die Modellierung von vager, unvollständiger Informationen. Bei der Modellierung mit RFA-Netzen müßten einige Sachverhalte erraten werden. Mit SeeMe hat man jetzt die Möglichkeit Lücken im Modell zu lassen, um sie später zu vervollständigen. Es ist z.B. aus den zur Verfügung stehenden Informationen nicht ersichtlich, welche Subaktivitäten die Aktivität „Begutachtung“ beinhaltet. Weiterhin ist es nicht möglich anzugeben, wel-

che Sub-aktivitäten der Aktivität "Betreuung" beinhaltet, da dies von Fall zu Fall unterschiedlich ausfällt. Bei der Modellierung mit der Methode SeeMe sollen zunächst die Basiselemente Rollen, Aktivitäten und Entitäten identifiziert werden. Anschließend können die Beziehungen zwischen den Elementen modelliert werden. Die Modellierung des Fallbeispiels "Prüfungsordnung Diplomarbeit" würde in SeeMe-Notation wie folgt aussehen:

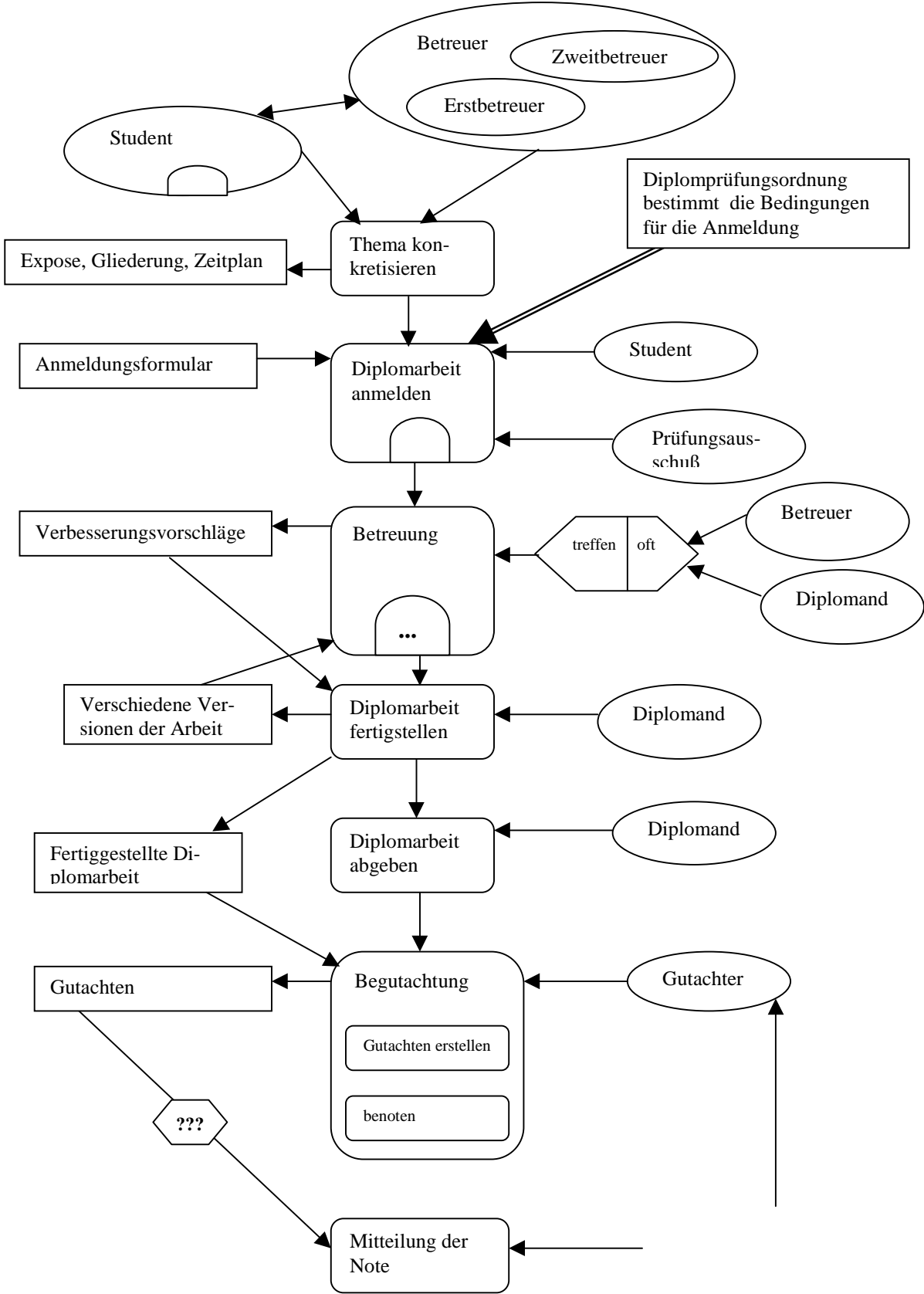




Abbildung 13: Modellierung des Fallbeispiels „Prüfungsleistung Diplomarbeit“ mit SeeMe

In der Abbildung 13 werden die Rollen, Aktivitäten, Entitäten und die Beziehungen zwischen diesen Elementen dargestellt. Die Verantwortlichkeiten von Rollenträgern lassen sich in SeeMe durch die Zuweisung der Rollenträger zu Aktivitäten beschreiben. Die Kooperation zwischen den Rollen kann im wesentlichen durch die gemeinsam verwendeten Gegenstände vergegenständlicht werden. In SeeMe kann das Rollenspiel bzw. Rollenwechsel leicht dargestellt werden. So tritt z.B. der Person, der eine Diplomarbeit schreiben will, als Student auf. Sobald seine Anmeldung bestätigt wird, tritt er im Modell in der Rolle "Diplomand" auf.

Im Unterschied zu RFA-Netzen kann mit der Methode SeeMe neben den Kontrollbeziehungen, auch andere Beziehungen beschreiben werden. So z.B ist möglich, sowohl die Beziehungen zwischen Aktivitäten und Entitäten als auch zwischen den Rollen untereinander zu modellieren. So z.B. konnte ausgedrückt werden, daß die Rollen "Student" und "Betreuer" Erwartungen voneinander haben. Der Student erwartet beispielsweise, daß er angemessen betreut wird. Der Betreuer hingegen erwartet vom Student, daß die Diplomarbeit gemäß seiner Erwartungen erstellt wird.

Es kann auch in SeeMe durch das Konzept Modifikator, die zweckdefinierten Ereignisse mit ihrer Wahrscheinlichkeiten modelliert werden. So konnte auch im Modell ausgedrückt werden, daß zum Zwecke der Durchführung der Aktivität "Betreuung", die Rollen "Student" und "Betreuer" sich öfters treffen können bzw. sollen.

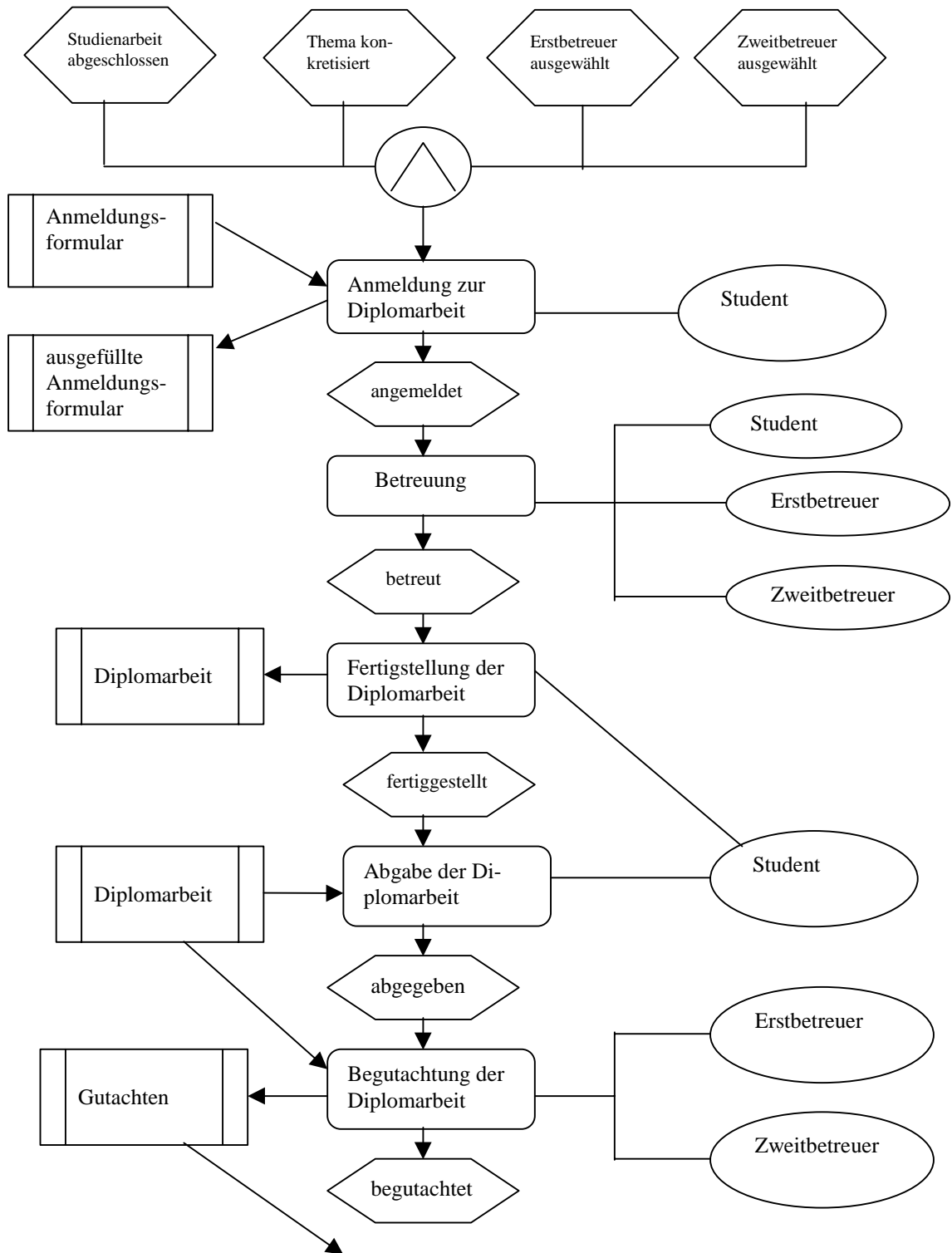
SeeMe bietet durch das Konzept MetaRelation die Möglichkeit die Beziehungen zwischen Elementen explizit zu beschreiben. Metarelationen in SeeMe werden entweder um die Semantik oder Struktur eines im Modell verwendeten Symbols auszudrücken oder um den Einfluß eines Elements auf die interne Struktur eines anderen Elements darzustellen, verwendet. In der Abbildung konnte durch dieses Konzept, der Einfluß der Entität "Diplomprüfungsordnung" auf die Aktivität "Diplomarbeit anmelden" zum Ausdruck gebracht werden.

Um die Übersichtlichkeit des erstellten Modells zu erhöhen, kann im SeeMe das Konzept der beabsichtigten Auslassung verwendet werden. In dem obigen Modell konnte z.B. zum Ausdruck gebracht werden, daß bewußt auf die Modellierung von Subaktivitäten der Aktivität „Anmeldung zur Diplomarbeit“ verzichtet wird.

Mit der Kennzeichnung unsicherer Informationen konnte im Modell darauf hingewiesen werden, daß die dem Modell zugrundeliegenden Informationen unvollständig oder unsicher sind. So war es möglich im Modell darauf hinzuweisen, daß die Aktivität "Betreuen" unvollständig spezifiziert ist. Damit wird auch darauf hingewiesen, daß der Modellierer nicht in der Lage ist, sie zu spezifizieren, da jeder Betreuung individuell abläuft. Im Modell wird an der Relation zwischen der Entität "Gutachten" und der Aktivität "Mitteilung der Note" gezweifelt. Der Zweifel hierbei besteht darin, ob die Note aus dem Gutachten entnommen wird, oder hierfür ein anderes Dokument existiert, worin die Note festgehalten ist.

4.2.3 Modellierung mit ARIS

Bei der Modellierung mit den erweiterten ereignisgesteuerten Prozeßketten können die Ereignisse, Funktionen, Organisationseinheiten und die Informations- bzw. Datenobjekte, die für die Durchführung von Funktionen benötigt werden, explizit modelliert werden. Die Modellierung des Prozesses "Diplomarbeit", mit den ereignisgesteuerten Prozeßketten, könnte beispielsweise wie folgt aussehen:



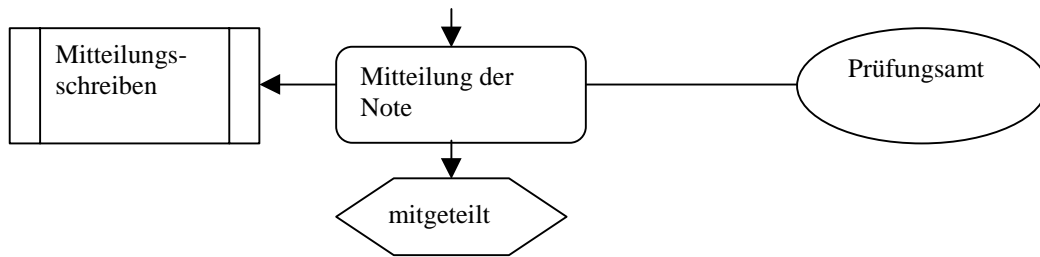


Abbildung 14: Modellierung des Fallbeispiels "Prüfungsleistung Diplomarbeit" mit eEPKs

Wie aus der Abbildung ersichtlich, wird der Prozeß "Diplomarbeit" als eine Kette von Funktionen modelliert, wobei Funktionen durch Ereignisse verknüpft sind. Die Funktion "Anmeldung zur Diplomarbeit" hat als Eingang die Startereignisse (Studienarbeit abgeschlossen, Thema konkretisiert, Erstbetreuer ausgewählt, Zweitbetreuer ausgewählt). In der Abbildung bedeutet dies, daß diese vier Ereignisse, bevor die Funktion "Anmeldung zur Diplomarbeit" ausgeführt werden kann, eingetreten sein müssen. Die Ausführung der Funktion "Anmeldung zur Diplomarbeit" löst wiederum das Ereignis "angemeldet" und dient als Startereignis für die Funktion „Betreuung“ usw. Mit dieser Verkettung wird der Kontrollfluß zwischen Funktionen modelliert.

Bei der Modellierung mit eEPKs fiel auf, daß die vier Bedingungen (Studienarbeit abgeschlossen usw.) im Vergleich zu RFA-Netzen und SeeMe leichter darzustellen sind. Mit der eEPK Methode ist nicht möglich, Verfeinerungen im Modell vorzunehmen. Dies kann durch die getrennte Modellierung in anderen Modellen vorgenommen werden.

4.2.4 Modellierung mit Action/Workflow

Bei der Modellierung mit diesem Ansatz muß zunächst eine Action/WorkflowLoop erstellt werden. Es werden bei der Erstellung von Modellen, die Interaktionen zwischen Menschen(gruppen) in den Mittelpunkt gestellt. Weiterhin muß der Prozeß in vier Phasen aufgeteilt werden. Da die Aussagekraft einer einzelnen Loop nicht sehr groß ist, können - die in den Phasen festgelegten Arbeitsabläufe in einem weiteren Modell (Action/Workflow Map) in seine Teilprozesse zerlegt werden. Ausgehend von dem Fallbeispiel "Diplomarbeit" müssen also zunächst die vier Bearbeitungsphasen festgelegt werden. Ebenfalls sollen die Rollen, die im Modell als Kunde und Dienstleister auftreten, identifiziert werden. Die Phasen und die zugehörigen Rollen und Sprechakte, die in dem primären Prozeß "Diplomarbeit" vorkommen, werden im folgenden zusammengefaßt.

| Phasen | Kunde | Sprechakt | Dienstleister |
|---|--|---|--|
| Phase 1: Themenvergabe | Student Arbeitsbereich Softwaretechnik | Kontaktaufnahme Themenbedarf erklären | Arbeitsbereich Softwaretechnik Student |
| Phase 2: Absprache der Bedingungen | Student Prüfungsausschuß | Thema präzisieren Diplomarbeit anmelden | Erstbetreuer und Zweitbetreuer Student |
| Phase 3: Anfertigung der Diplomarbeit | Erstbetreuer und Zweitbetreuer Student | Teilergebnisse vorstellen und absprechen Lesen bzw. korrigieren der Teilergebnisse | Diplomand Erstbetreuer und Zweitbetreuer |
| Phase 4: Begutachtung | Prüfungsamt Prüfungsamt | Gutachten erstellen Diplomarbeit benoten | Erstbetreuer und Zweitbetreuer Erstbetreuer und Zweitbetreuer |

Tabelle 6: Sprechakte im Fallbeispiel "Prüfungsleistung Diplomarbeit"

Mit diesem Ansatz kann der gesamte Prozeß, von der Themenvergabe bis zur Betreuung bzw. Benotung, in einem Modell übersichtlich dargestellt werden. Im Mittelpunkt des Modells steht der Hauptprozeß Diplomarbeit und die beiden (Haupt) Rollen Student und Betreuer. Die Teilprozesse werden getrennt in einem anderen Modell dargestellt.

Die graphische Darstellung der Phasen und Rollen im primären Prozeß würde mit dem Action/Workflow- Ansatz folgendermaßen aussehen:

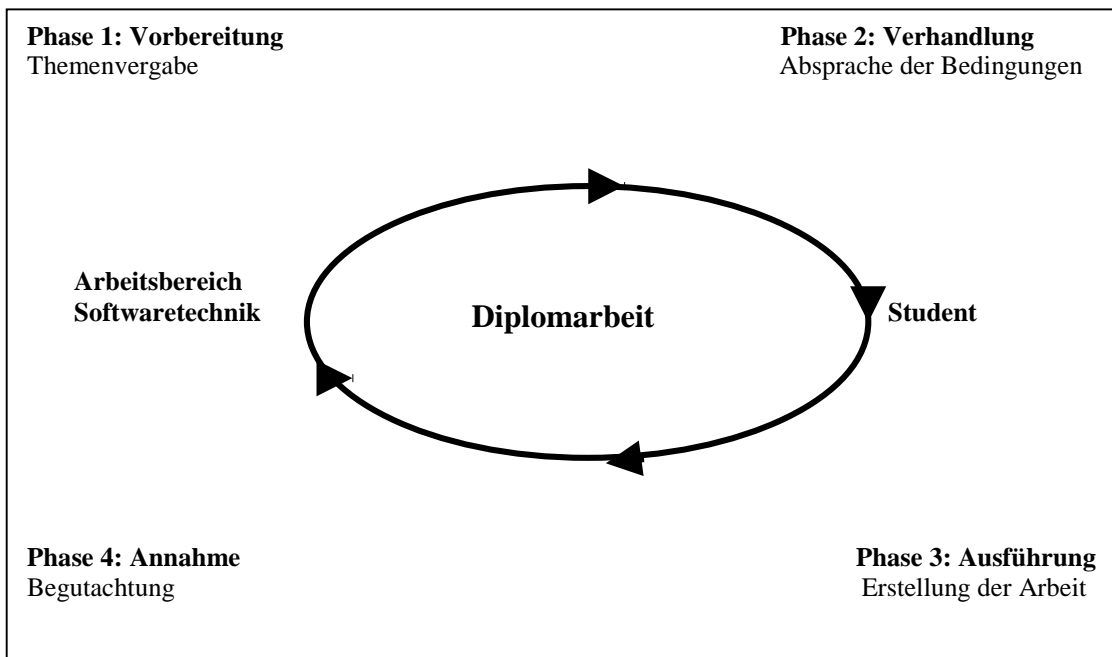


Abbildung 15: Fallbeispiel "Prüfungsleistung Diplomarbeit" aufgeteilt in Phasen und Rollen

In diesem Modell tritt der Arbeitsbereich Softwaretechnik in der Rolle des Kunden und der Student in der Rolle des Dienstleisters auf. In der ersten Phase des Prozesses (Vorbereitungsphase) bietet Arbeitsbereich Softwaretechnik Themen für die Diplomarbeiten (in den Projektseminaren) zur Bearbeitung an. In der Verhandlungsphase werden dann über die Bedingungen, die für die Fertigstellung der Arbeit notwendig sind, abgesprochen. Hierbei muß die Arbeit präzisiert und angemeldet werden. In der dritten Phase des Prozesses "Ausführungsphase", stellt der Student unter der Beachtung der abgesprochenen Bedingungen die Diplomarbeit fertig. Anschließend in der "Annahmephase" überprüft der Betreuer die erstellte Leistung (Diplomarbeit) und bewertet sie. Dieser Phase wird dann abgeschlossen betrachtet, wenn der Betreuer die Arbeit zufriedenstellend findet. Sonst muß die Schleife von vorne durchlaufen bzw. die Phase 3 "Erstellung der Diplomarbeit" wiederholt werden.

Da die Aussagekraft einer Workflow Loop nicht sehr groß ist, besteht in diesem Ansatz weiterhin die Möglichkeit diesen Prozeß zu verfeinern, indem die Teilprozesse mitaufgeführt werden. Die sekundären Prozesse können untereinander verkettet werden. Hiermit bekommt man einen Überblick über die Konversationen, die für den Prozeß wichtig sind.

In der folgenden Abbildung wird Action/Workflow Map des Prozesses "Diplomarbeit" mit seinen sekundären Prozessen dargestellt. Es wird hierbei wegen der Übersichtlichkeit nicht alle Teilprozesse aufgeführt

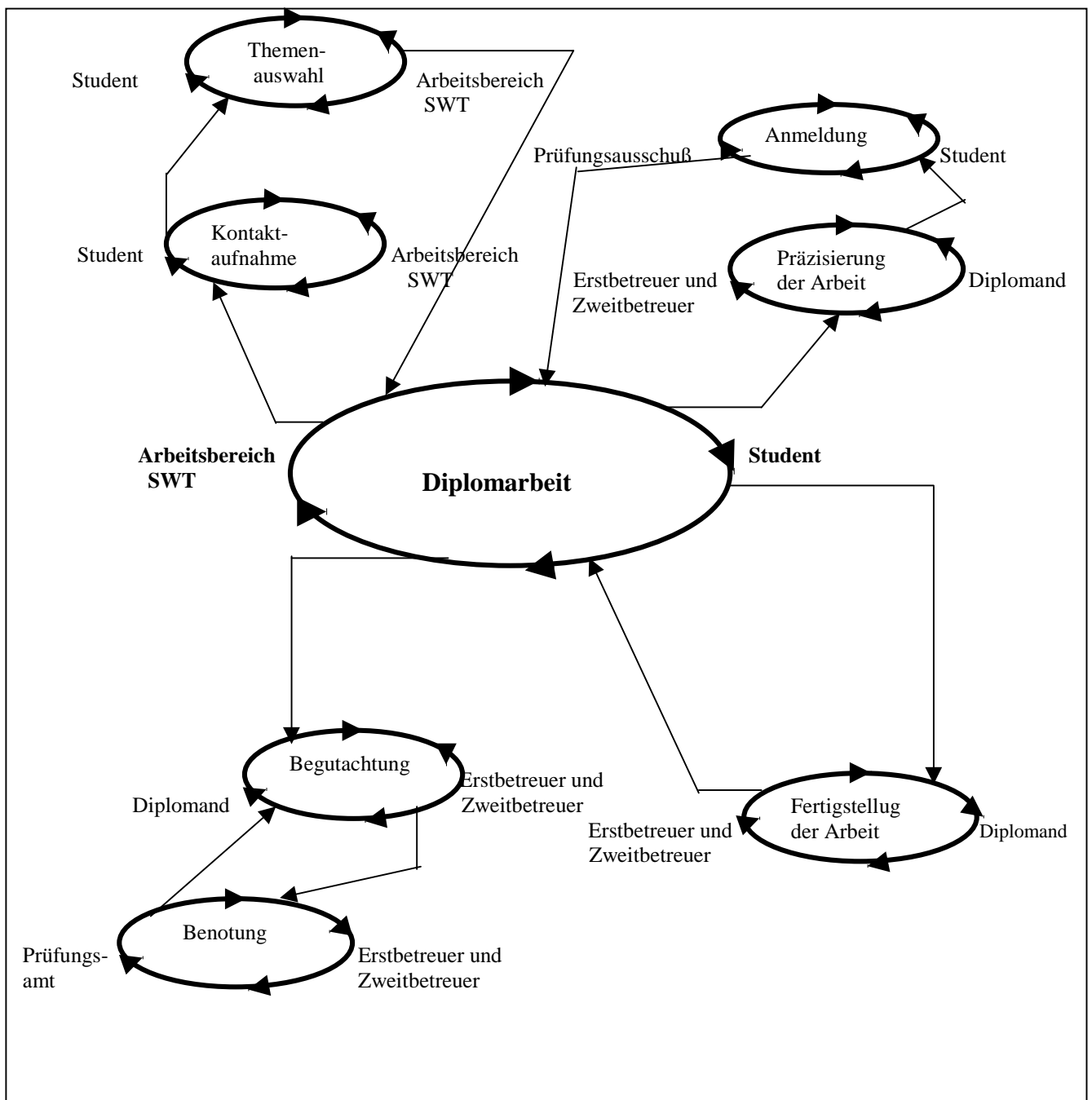


Abbildung 16: Modellierung des Fallbeispiels "Prüfungsleistung Diplomarbeit" mit Action/Workflow

Wie schon aus der Abbildung hervorgeht, stellt der primäre Prozeß "Diplomarbeit" den Ausgangspunkt für die weiteren Teilprozesse. Innerhalb jeder Phase des primären Prozesses können mehrere sekundäre Prozesse ausgelöst werden. Es wird in der Abbildung vereinfacht die Teilprozesse in den einzelnen Phasen abgebildet. In den Teilprozessen wechselt teilweise das Kunde/Dienstleisterverhältnis. Beispielsweise, bei dem Teilprozeß „Themenauswahl“ tritt „der Student“ in der ersten Phase als Kunde und „der Betreuer“ als Dienstleister auf.

4.3 Vergleich der Ansätze

In diesem Teil der Arbeit sollen nun die Ansätze im Hinblick auf die Erstellung komplexer Modelle verglichen werden. Es sei jedoch darauf hingewiesen, da dieser Vergleich überwiegend auf die subjektiven Erfahrungen basiert, daher als bedingt aufzufassen ist.

Die Rollen-/Funktions-/Aktions-Netze eignen sich insbesondere bei der Modellierung von Arbeitsprozessen, wo der Austausch von Dokumenten bzw. Objekten im Vordergrund stehen. Die Schwierigkeit bei der Erstellung der Modelle bestand insbesondere darin, daß die RFA-Netze schwer erlernbar sind. Dies ist jedoch auf ihre Ausdrucksmächtigkeit zurückzuführen. RFA-Netze weisen eine hohe Ausdrucksmächtigkeit auf, insbesondere dort, wenn es um die Modellierung von Objekten geht. Die Objekte können bis ins Detail modelliert werden. Es ist positiv zu bewerten, daß für die Erstellung eines Modells nicht alle Bestandteile der Sprache erlernt werden müssen. Die statische Beschreibung von Aufgaben ist mit einfachen Elementen (Rollen und Funktionen) möglich. Wenn jedoch die Aktionen modelliert werden sollen und damit der volle Umfang der RFA-Netze erlernt werden muß, scheint dies sehr aufwendig zu sein. Mit RFA-Netzen war es schwierig, die kommunikationsintensiven Teile des Prozesses zu modellieren. Es kann jedoch im Modell explizit auf die Art von Kooperation - gemeinsamer Handlungen oder über Schnittstellen - hingewiesen werden. Durch die Modellierung Objekt- und Kontrollaspekt hat man die Möglichkeit - falls nötig - diese Aspekte getrennt zu modellieren. Hiermit kann man sich auf die einzelnen Aspekte konzentrieren. Mit den Aktionsnetzen konnten dann alle Aspekte zusammen betrachtet werden. Es sei jedoch an dieser Stelle darauf hingewiesen, daß in den obigen Modellen benutzten Basiselemente nur ein Teil von RFA-Notation darstellen. Es besteht beispielsweise die Möglichkeit im Modell die automatisierten Anteile der Aktivitäten mitzumodellieren.

SeeMe unterscheidet sich von den anderen drei ausgewählten Ansätzen dadurch, daß die Beziehungen der Elemente nicht ausschließlich auf die Kontroll- bzw. Objektfluß beschränkt ist. Mit dieser Methode können weitere Beziehungen modelliert bzw. neue hinzugefügt werden. Probleme in diesem Zusammenhang bestand darin, diese vielfältigen Relationen zwischen den Elementen zu verstehen. Die Standardbedeutung von Relationen zwischen den einzelnen Basiselementen ist gewöhnungsbedürftig. Dies kann als eine Schwäche in bezug auf die Erlernbarkeit der SeeMe aufgefaßt werden. Wenn jedoch sie einmal verstanden sind, kann mit der Methode SeeMe viel im Modell - insbesondere wenn es um soziale Aspekte geht - ausgedrückt werden. Es kann z.B. durch Hinzufügen von Attributen, arbeitswissenschaftliche Kriterien - wie KABA-Verfahren- bei der Modellierung mitberücksichtigt werden. Da alle Aspekte in einem Modell dargestellt werden müssen, kann hierdurch die Lesbarkeit der Modelle gesenkt werden. Um die Lesbarkeit zu erhöhen, kann das Konzept "beabsichtigter Auslassung" herangezogen werden. Dem Modellierer wird damit Möglichkeit gegeben, die Sachverhalte je nach Modellierungszweck wegzulassen und im Modell zu vermerken. Damit hat man die Möglichkeit, komplexe Zusammenhänge zunächst vage zu modellieren und erst bei Bedarf zu verfeinern. Es war jedoch nicht einfach dieses Konzept praktisch anzuwenden, da die Auswahl der Symbole einen erheblichen Aufwand bei der Modellierung stellte. Dieses Konzept kann eine Hilfestellung bieten, wenn z.B. bei der Analyse nicht die Möglichkeit besteht, die einzelnen Personen zu befragen, oder wenn die Personen ihre Arbeit und die Kooperation mit den anderen nicht genau beschreiben können.

Die Schwierigkeit bei der Erstellung komplexer Modelle mit ARIS besteht darin, daß man sich vor Modellierungsbeginn darüber Gedanken machen muß, was dargestellt werden soll, da durch die fünf Sichten eigene Methoden zum Einsatz kommen. Die Erstellung des Modells für das Fallbeispiel mit der eEPKs war nicht so schwierig. Dies kann damit begründet werden, daß für die Modellierung lediglich die Identifikation der Funktionen, Ereignisse, Organisationseinheiten und Informationsobjekte ausreicht. Die Modellierung von Prozessen mit Hilfe von eEPKs ist insbesondere dann geeignet, wenn die Prozeßabläufe im voraus bekannt sind. Wenn jedoch der Ablauf der Aufgabenerfüllung ad-hoc und nicht klar ist - wie z.B. das Betreuungsverhältnis zwischen dem Student und Betreuer - dann ist die Darstellung dieser Abläufe sehr schwer bzw. nicht möglich. Solche Fälle sind mit dem Action/Workflow Ansatz besser handzuhaben.

Bei der Modellierung mit dem Action/Workflow Ansatz fiel auf, daß die Identifizierung des Kunde/Dienstleistungsverhältnisses, und die Aufteilung des Modells in Phasen nicht einfach ist. Da der Mensch und sein Handeln bei der Modellierung im Vordergrund gestellt ist, können die

kommunikationsintensiven Arbeitsabläufe mit diesem Ansatz - im Vergleich zu den anderen drei Ansätzen - leichter modelliert werden. Um einen Action/Workflow-Modell zu erstellen, müssen die Konversationen, die zwischen Menschen stattfinden, identifiziert und modelliert werden. Hiermit unterstellt jedoch dieser Ansatz, daß jeder Arbeitsprozeß sich auf Konversationen zurückführen läßt. Es ist in diesem Zusammenhang auch negativ zu bewerten, daß die Ressourcen (Objekte, Dokumente), die zur Durchführung von Arbeit benötigt werden, nicht im Modell explizit gemacht werden können. Die erstellten Modelle können mit Hilfe des Werkzeugs Action/Application Builder zur Entwicklung von Groupwareanwendungen herangezogen werden.

Bei diesen vier Ansätzen ist der Syntax und Semantik der Darstellungselemente von der Modellierungsmethode fest vorgegeben. Der Modellierer hat hierbei keine Freiheit in der Modellierungssprache, bei Bedarf, neue Elemente hinzuzufügen. Die Modellierungsmethode kann - wenn dann - nur vom Entwickler der Methode erweitert werden. Es wäre jedoch wünschenswert, daß der Modellierer selbst die Modellierungsnotation erweitern bzw. an ihre Bedürfnisse anpassen kann.

Diese Flexibilität bei der Modellierung wird von dem Geschäftsprozeßmanagementwerkzeug ADONIS (Analyse, Design, Optimierung natürlich inklusive Simulation) angeboten [BOC99]. Es basiert, die von Business Process Management System (BPMS) Gruppe der Universität Wien erarbeiteten Forschungsergebnissen auf und wird von der BOC GmbH vertrieben. Es scheint an dieser Stelle der Arbeit sinnvoll, diesen Ansatz näher aufzuführen, da sie eine andere Herangehensweise an die Modellierung aufweist, als für diese Arbeit ausgewählten Ansätzen, uns sich auch mit der Softwareentwicklung beschäftigt.

Der Unterschied besteht darin, daß ADONIS beliebige Modellierungstechniken, die nicht auf die Prozeßmodellierung beschränkt sein müssen zu definieren erlaubt. Es bedeutet, daß die Metamodellierungskonzepte von dem Modellierer frei konfigurierbar und erweiterbar sind. ADONIS bietet die Möglichkeit, vor dem Einsatz des Werkzeuges ein Customizing durchzuführen. Im Rahmen der Customizing von ADONIS, kann die gewünschte Modellierungsmethode ohne Programmieraufwand frei definiert werden. ADONIS nutzt dazu ein objektorientiertes Metamodellierungskonzept. Es kann frei festgelegt werden, welche Modellierungselemente man benutzen will, wie diese graphisch aussehen sollen und welche Attribute diese besitzen sollen. Dem Modellierer wird hierdurch die Möglichkeit gegeben, die Modellierungsnotation und Modellsichten frei zu gestalten und zu konfigurieren. Somit ermöglicht dieses Werkzeug, sowohl verschiedene exi-

stierender Modellierungstechniken zu simulieren wie z.B. ARIS oder UML, als auch neue zu definieren.

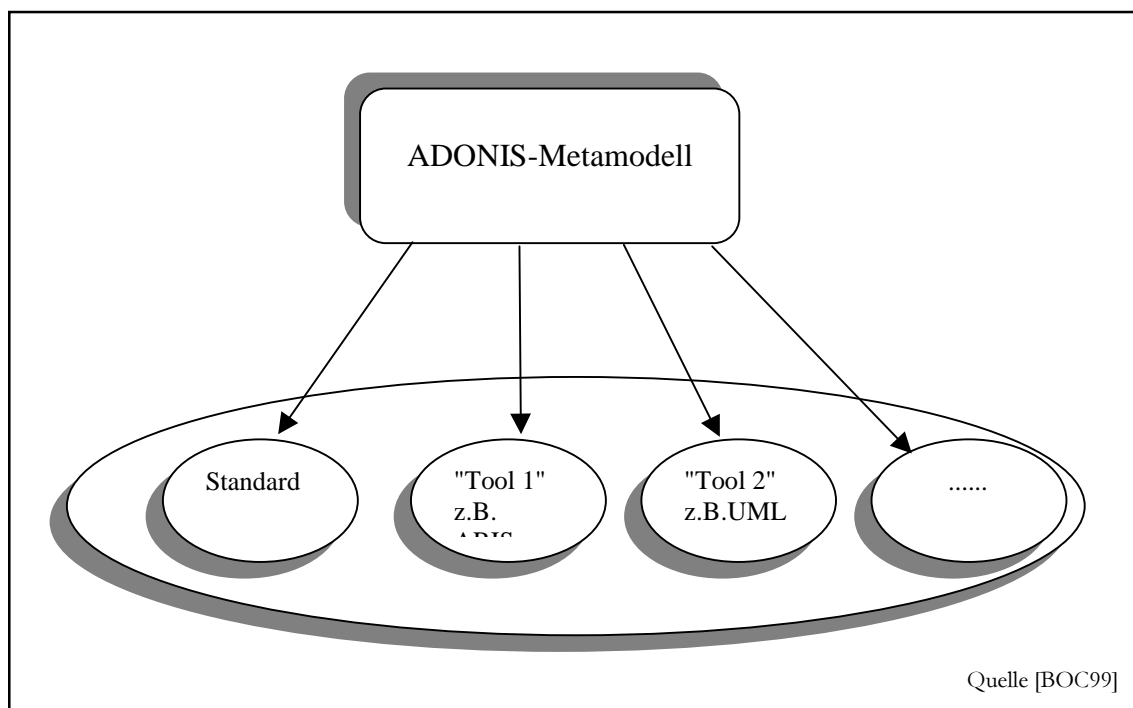


Abbildung 17: ADONIS-Customizing

Hierzu bietet das Werkzeug ADONIS zwei Migrations-Mechanismen (Online und Offline). Beim Online-Mechanismus können während des laufenden Betriebs von ADONIS - während der Arbeit des Modellierers - die Modellsichten und Modellierungsnotation angepaßt oder geändert werden. Der Offline-Mechanismus unterstützt den Metamodellierer bei den strukturellen Änderungen einer Modellierungstechnik. Beispielsweise können Klassen, Beziehungstypen und Attribute neu definiert oder aus bestehenden Modellierungstechniken entfernt werden.

Nachdem in diesem Kapitel ein praktischer Vergleich der Ansätze durchgeführt ist, wird im folgenden die Ansätze anhand von Anforderungen aus den verschiedenen Sichtweisen - die im Kapitel 2 herausgearbeitet worden sind - bewertet.

Kapitel 5

Ergebnisse/Fazit

5.1 Bewertung der Ansätze zur Kooperationsmodellierung

In diesem Teil der Arbeit werden die Ansätze anhand von Anforderungen, die im Kapitel 2.6 herausgearbeitet worden sind, dahingehend bewertet, ob und wie die Ansätze die Anforderungen aus verschiedenen Sichtweisen der Kooperationsmodellierung erfüllen. Es sei jedoch an dieser Stelle darauf hingewiesen, daß dieser Bewertung als eine bedingte Bewertung aufgefaßt werden soll. Erstens hat diese Bewertung keine empirische Grundlage und zweitens unterscheiden sich die Anforderungen wesentlich voneinander - da sie aus verschiedenen Perspektiven gestellt werden. Dies hat jedoch den Vorteil, daß die Bewertung nicht einseitig erfolgt.

1. Erstellung von Ist- und Sollmodellen

RFA-Netze: Mit RFA-Netzen können Ist- und Sollmodelle erstellt werden. In den Modellen können die Aktivitäten, Funktionen, Rollen und Objekte abgebildet und ihre Zusammenhänge verdeutlicht werden. Aus organisationsorientierter Sichtweise ist jedoch zu bemängeln, daß die Aspekte aus der Sicht einer Person betrachtet werden. Dies führt dazu, daß die abteilungsübergreifenden Zusammenhänge nicht leicht zu verstehen, zu erkennen und zu optimieren sind.

SeeMe: Mit SeeMe können ebenfalls Ist- und Sollmodelle erstellt werden. Die Modelle können auch zur Schwachstellenanalyse herangezogen werden. Da bei SeeMe bei der Modellierung mitarbeiterorientierte Aspekte mitbetrachtet werden, können in Modellen neben organisatorischen Schwachstellen auch die Schwachstellen in bezug auf die Belastung der Mitarbeiter etc. entdeckt werden.

ARIS: Mit ARIS können Ist- Sollmodelle erstellt werden, die eine Abbildung betrieblicher Prozesse ermöglicht. Mit ARIS können die Geschäftsprozesse über Abteilungsgrenzen hinweg, in eEPK-Modellen dargestellt werden. Die erstellten Modelle können mit dem Analyse- und Modellierungswerkzeug ARIS-Toolset analysiert und bezüglich der Kosten und Zeiten bewertet werden. Damit können Schwachstellen und Verbesserungspotentiale in dem Geschäftsprozeß aufgezeigt werden.

Action/Workflow: Die Erstellung von Ist- und Sollmodellen ist mit diesem Ansatz möglich. Aus dem Action/Workflow-Modell können die Verantwortlichen eines Geschäftsprozesses abgeleitet werden. Anhand des Modells kann festgestellt werden, wer für welche Tätigkeiten innerhalb eines Geschäftsprozesses zuständig sind. Das Modell unterstützt die Optimierung von Geschäftsprozessen, indem es aufzeigt, wo Tätigkeiten parallel ausgeführt werden. So können die Schwachstellen in dem Prozeß erkannt werden. Anhand von Erfüllungsbedingungen kann auch die Belastung der Mitarbeiter im Modell sichtbar gemacht werden. Aus organisatorischer Sichtweise heraus zu bemängeln, daß durch die starke Betonung der Interaktion zwischen Menschen, die Zusammenhänge und die Reihenfolge der Aktivitäten aus dem Modell nicht leicht zu entnehmen sind.

Expliziter Hinweis auf die Phänomene der Arbeitspraxis

RFA-Netze: In RFA-Netzen wird die Arbeit aus der Perspektive einer Person betrachtet. Dabei wird die Gesamttätigkeiten einer Person als Handeln in verschiedenen Rollen aufgefaßt, die in einem Netzwerk kooperierender Rollen eingebettet sind. Hiermit wird darauf hingedeutet, daß die Arbeit aus der Perspektive der handelnden Menschen verstanden werden soll. Aus arbeitsorientierter Sichtweise ist jedoch zu bemängeln, daß dabei nicht die Aspekte, die nicht in den Modellen abgebildet werden können, mit RFA-Netzen nicht genügend thematisiert werden können.

ARIS: ARIS betrachtet durch seine Orientierung auf die Modellierung von Geschäftsprozessen, die Arbeit aus der Perspektive einer Organisation. Somit unterstellt sie auch, daß die Arbeit und die situierte menschliche Handlung durch die Modellierung von Geschäftsprozessen abgedeckt werden können. Die Modellvorstellung ist dabei, daß die menschliche Handlung durch die Steuerung – wie der Name Steuerungssicht in ARIS andeutet - der Abläufe handhabbar ist. Den Interessenkonflikten, Motivationsaspekten, Entscheidungsspielräumen etc. werden weder in der Modellierungssprache noch in der Vorgehensweise Rechnung getragen. Dies bestätigen auch die Probleme, die mit der Einführung von Standardsoftware R/3 System in der Praxis entstehen. Die Mitarbeiter finden sich dadurch in ihrer Arbeit nicht wieder, die Kooperation wird erschwert etc. Von Einsetzung von Kommunikationsmethoden oder ethnographische Methoden, um Arbeitspraxis zu verstehen, ist nicht die Rede. Es wird gehofft, daß durch die Schulung eine angemessene Benutzung des Systems ermöglicht wird.

SeeMe: Da bei der Entwicklung der SeeMe-Methode die Modellierung der Kommunikationsstrukturen, Erwartungen und Interessen eine große Rolle gespielt haben, wird die situierte menschliche Handlung von Anfang an bei der Modellierung thematisiert. Der Versuch, die Humankriterien des KABA-Verfahrens in die Modellierung miteinzubeziehen und die Ergebnisse ethnographischer Methoden in die Modelle aufzunehmen, ist aus arbeitsorientierter Sichtweise positiv zu bewerten.

Action/Workflow: Dieser Ansatz weist - durch die Modellierbarkeit von Zielen, Erwartungen, Wünsche, Kommunikation etc. in Form von Erfüllungsbedingungen und der Konversationen zwischen den Menschen(gruppen)- im Modell explizit auf die Phänomene der Arbeitspraxis hin. Da die Interaktion zwischen Menschen in den Mittelpunkt gestellt wird, rückt bei der Analyse und Modellierung die menschliche Handlung und Kommunikation in den Mittelpunkt der Interesse. Die Abhängigkeit der Arbeitsabläufe von den Erfüllungsbedingungen deutet auch darauf hin, daß die Entscheidung über den Ablauf des Prozesses bei den Menschen bleibt. Weiterhin ist

aus arbeitsorientierter Sichtweise ist positiv zu bewerten, daß der Ausführung des Prozesses erst dann erfolgt, wenn das Ergebnis zufriedenstellend ist. Es kann jedoch in diesem Zusammenhang kritisiert werden, daß die Kommunikationsbeziehungen in Organisationen nicht auf die Sprechakte reduzieren läßt und die Beziehungen sich nicht immer mit den vier Phasen beschreiben lassen.

Implementierbarkeit der Modelle

RFA-Netze: Da die RFA-Netze die Präzisierung der Objekttypen und der automatisierten Funktionen bis hin zu einer vollständigen Formalbeschreibung ermöglichen, können aus RFA-Modelle Vorgaben für die Programmierung gewonnen werden. Es ist mir leider nicht bekannt, ob auf der Basis dieser Netze Systeme implementiert worden sind. Die Modelle könnten jedoch prinzipiell mit Workflow-Management-Systemen, die diese Art von Petri-Netzen (gefärbte Netze) unterstützen, implementiert werden.

SeeMe: Bei der Entwicklung von SeeMe spielte die Kommunizierbarkeit der Modelle und die Aufnahme von sozialen Aspekten in Modellen eine große Rolle. Die Implementierbarkeit der Modelle stand nicht im Vordergrund. Die Modelle enthalten daher viele Aspekte, die nicht formalisiert werden können und daher auch nicht zur Implementierung zugänglich sind. Da SeeMe eine neue Methode ist, ist abzuwarten, ob in Zukunft, in bezug auf die Implementierbarkeit der SeeMe-Modelle gearbeitet wird.

ARIS: Die eEPK-Modelle können an sich nicht implementiert werden. Es gibt jedoch einige kommerziell erhältliche Workflow-Management-Systeme (z.B. COSA), die sich anstrengen, die eEPK-Modelle automatisch auszuführen. Zur Zeit sieht es jedoch so aus, daß die eEPK-Modelle noch nicht ausgeführt werden können. Es kann lediglich eine Übersetzung in die Prozeßdefinitionen einiger Workflow-Management-Systeme erfolgen.

Action/Workflow: Die Action/Workflow Modelle können mit dem Action/Workflow Application Builder definiert und ausgeführt werden. Es steht auch eine Anwendungsprogrammierschnittstelle zur Verfügung, welche die Anbindung der Anwendungen an das Action/Workflow System ermöglichen. Es ist hierbei möglich, mit Hilfe geeigneter Software (wie z.B. Visual C++) Formulare und Sichten zu programmieren.

Werkzeugunterstützung:

RFA-Netze: Speziell für die RFA-Netze existiert keine Werkzeugunterstützung. Die Modelle können mit normalen Editoren und mit Petri-Netz Werkzeugen erstellt werden.

SeeMe: Für die Erstellung von SeeMe Modellen ist zur Zeit kein Werkzeug verfügbar. Es wird an der Entwicklung eines Editors namens "EasySeeMe" gearbeitet, die die Erstellung der SeeMe-Modelle erleichtern soll.

ARIS: Für die Erstellung und Analyse von Modellen steht das Analyse- und Modellierungswerkzeug ARIS-Toolset. Mit diesem Werkzeug können die betrieblichen Abläufe analysiert und bzgl. Kosten und Zeiten bewertet werden. Weiterhin können Ist- und Sollmodelle generiert und verglichen werden. Die Modellierungskomponente ist entsprechend des ARIS-Konzeptes nach Beschreibungssichten und Ebenen aufgebaut. In jedem Sicht kommt eine andere Modellierungsmethode zum Einsatz. Die Verknüpfung der Modelle erfolgt dann über Steuerungssicht.

Action/Workflow: Die Erstellung der Action/Workflow Modelle werden durch das Werkzeug Action/Workflow Analyst unterstützt. Neben der Erzeugung der Action/Workflow Maps, bietet dieses Werkzeug die automatische Berechnung von Durchlaufzeiten und Kosten der Prozesse an, die in „conditions of satisfaction“ festgelegt werden können. Die erzeugten Daten können in tabellarischer Form in andere Anwendungen (z.B. Tabellenkalkulation) exportiert werden. Das Werkzeug Action/Workflow Application Builder kann die erstellten Modelle automatisch ausführen. Dieses Werkzeug erzeugt Workflow-Spezifikationen, welche in einer Datenbank verwaltet werden. Diese Datenbank wird von Action/Workflow Manager als Grundlage für die Ausführung und Steuerung von Workflows benutzt.

5.2 Zusammenfassung und Ausblick

Gegenstand dieser Arbeit war der Vergleich der Ansätze zur Kooperationsmodellierung. Der Kooperationsmodellierung wurde als "*die Ansätze im Rahmen der Anwendungsentwicklung, die im Modell wesentliche Aspekte von Kooperation sowohl visuell als auch konzeptionell explizit machen*", definiert. Im Hinblick auf diese Definition werden vier Ansätze zur Kooperationsmodellierung ausgewählt, sowohl analytisch als auch praktisch verglichen. Um eine systematische Bewertung der Ansätze

durchführen zu können, sind - ausgehend aus der Problemstellung und aus Beobachtungen in den Projekten - vier Sichtweisen der Kooperationsmodellierung herausgearbeitet. Diese Anforderungen sind dann zur Bewertung der Ansätze herangezogen.

Die erste Fragestellung der Arbeit *"welche Aspekte von Kooperation werden im Modell sowohl konzeptionell als auch visuell explizit gemacht?"* ist durch den analytischen und praktischen Vergleich der Ansätze beantwortet. Der analytische Vergleich ist nach einem vordefinierten Analyseraster durchgeführt. Es sind dabei zwischen vier Kategorien der Analyse unterschieden:

1. Entstehungsgeschichte
2. Theoretischer Fundierung
3. Basiselemente und ihre Verknüpfung
4. Einsatzkontext.

Der analytische Vergleich hat gezeigt, daß die Ansätze sich voneinander wesentlich unterscheiden. Diese Unterschiede beziehen sich einerseits auf die Einsatzgebiete andererseits auf die Theorien, auf denen die Ansätze fundieren. Auch die Basiselemente und ihre Verknüpfungen unterscheiden sich wesentlich voneinander. Während die Modellierungsnotation der RFA-Netze, SeeMe und ARIS einige Gemeinsamkeiten aufweisen, unterscheidet sich der Ansatz Action/Workflow visuell wesentlich von den anderen drei Ansätzen.

Der praktische Vergleich, der anhand des Fallbeispiels „Prüfungsleistung Diplomarbeit“ durchgeführt ist, bringt das Zusammenspiel der Basiselemente und ihre Verknüpfungen zum Vorschein. Die Erfahrungen, die bei der Erstellung der Modelle gemacht sind, werden dann für den praktischen Vergleich herangezogen.

Die Antwort auf die zweite Fragestellung der Arbeit *"Wie unterstützen diese Ansätze die Integration zwischen Softwareentwicklung einerseits Organisationsentwicklung und Arbeitsgestaltung andererseits"* ist im wesentlichen durch die Bewertung der Ansätze aus verschiedenen Sichtweisen der Kooperationsmodellierung beantwortet.

Da die RFA-Netze auf die Petri-Netze basieren, bietet sie eine gute Unterstützung bei der Softwareentwicklung. Sie stößt jedoch dort an Grenzen – obwohl sie im Modell explizit auf die Aspekte von Kooperation (Kontrollfluß, Kooperations- und Interaktionsbeziehungen, Rollen)

aufweist -, wenn die Phänomene der Arbeitspraxis bei der Modellierung näher betrachtet werden sollen. SeeMe beschäftigt sich hingegen mit diesen Themen explizit und versucht die sozialen Aspekte in die Modellierung miteinzubeziehen. Sie stößt jedoch dort an Grenzen, wenn es um die Softwareentwicklung geht. ARIS beschränkt sich auf die Geschäftsprozeßmodellierung und verliert dadurch die Arbeitspraxis und bietet keine angemessene Unterstützung für die Softwareentwicklung. Der Ansatz Action/Workflow hingegen, stößt dort an Grenzen, wenn es um die Modellierung von Objekten geht. Action/Workflow-Ansatz geht davon aus, daß die kooperativen Arbeitsprozesse mit Hilfe der Sprechakttheorie formal ontologisch untersucht werden können. Mit der Phaseneinteilung unterstellt sie, daß die Kooperationsbeziehungen sich mit den vier Phasen beschreiben lassen.

Dieser Vergleich hat – auch wenn sie spekulativ und nicht empirisch fundiert ist - gezeigt, daß mit keinem der ausgewählten Ansätze die Integration zwischen Softwareentwicklung einerseits Organisationsentwicklung und Arbeitsgestaltung andererseits, angemessen zu bewältigen ist. Der Ansatz Action/Workflow könnte im Vergleich zu den anderen Ansätzen hierfür eine gute Unterstützung anbieten. Was jedoch in diesem Zusammenhang zu bemängeln ist, daß in diesem Ansatz die Objekte und Dokumente nicht im Modell explizit gemacht werden können. Da die RFA-Netze eine sehr hohe Ausdrucksmächtigkeit aufweisen und für die Softwareentwicklung eine gute Unterstützung anbieten können, kann für zukünftige Forschungsarbeiten von Interesse sein, diese Netze im Hinblick auf die Organisationsentwicklung und Arbeitsgestaltung zu erweitern.

Die SoftwareentwicklerInnen können in ihrer Arbeit heutzutage nicht auf die UML verzichten. Aus den Ergebnissen dieser Arbeit heraus, sehe ich daher einen Forschungsbedarf, im Hinblick auf die Integration zwischen UML und Kooperationsmodellierung.

Literaturverzeichnis

- **[And97]**
Urs Andelfinger:
„Diskursive Anforderungsanalyse
Ein Beitrag zum Reduktionsproblem bei Systementwicklungen in der Informatik“
Peter Lang Verlag. Frankfurt a. M. (1997)
- **[Ban95]**
Liam J. Bannon:
"The Politics of Design: Representing Work"
Communication of the ACM 38 (1995) 9, S.66-68
- **[BBH89]**
P. Hennessy, S. Benford, S., and J. Bowers:
„Modeling group communication structures: analysing four European projects“
In Proceeding of ECSCW '89, London UK, (1989), S. 406 420
- **[BEH99]**
Keith A. Butler, Chris Esposito, Ron Hebron:
"Connecting the Design of Software to the Design of Work"
Communication of the ACM 42 (1999) 1, S. 38 - 46
- **[BHK94a]**
G. Barkow, W. Hesse, H. -B. Kittlaus, G. Scheschonk, H.von Braun
"Terminologie der Softwaretechnik.
Ein Begriffssystem für die Analyse und Modellierung von Anwendungssystemen

Teil 1: Begriffssystematik und Grundbegriffe.“

Informatik Spektrum 17, (1994) S. 39 - 47

- **[BHK94b]**

G. Barkow, W. Hesse, H. -B. Kittlaus, G. Scheschonk, H.von Braun

"Ein Begriffssystem für die Analyse und Modellierung von Anwendungssystemen

Teil 2: Tätigkeits- und ergebnisbezogene Elemente"

Informatik Spektrum 17 (1994), S. 96 - 105

- **[BHÖ95]**

Leo Brecht, Thomas Hess, Hubert Österle:

"Stand und Defizite der Methoden des Business Process Redesign"

Wirtschaftsinformatik 37 (1995) 5. S. 480 - 486

- **[BJ97]**

Rainer Burkhardt, Ivar Jacobon:

"Interview - Im Gespräch "

Objektsprktrum 3(1997), S. 20-22

- **[BMS95]**

K. Bauknecht, T. Mühlherr, C. Sauter, S. Teufel:

„Computerunterstützung für die Gruppenarbeit“

Addison Wesley (1995)

- **[BOC99]**

ADONIS® -Komponenten: Modellierung:

<http://www.boc-eu.com> (August99)

- **[BP94]**

B. Büchel, G. Probst:

"Organisationales Lernen"

Wiesbaden, Gabler (1994)

- **[BR91]**

Liam Bannon, Mike Robinson:

„ Questioning representation“

Proceedings of ECSCW '91 (1991), S. 219 – 233

- **[BS89]**

Liam Bannon, Kjeld Schmidt:

„CSCW: Four Characters in Search of a Context“

Proceedings of the First European Conference on Computer Supported Cooperative Work, ECSCW'89 (1989) London

- **[CK99]**

John M. Carroll, Hermann Kaindl:

Symbolic Modeling in Practice

Communication of the ACM 42 (1999) 1, S. 28 -30

- **[DDJ98]**

G. De Michelis, E. Dubios, M. Jarke, F. Matthes, J. Mylopoulos, J. W. Schmidt, J., C. Woo, E. Yu

„A tree faceted view of information systems“

Communication of the ACM 41 (1998) 12, S. 64 – 70

- **[DHB99]**

B. Deifel, U. Hinkel, B. Paech, P. Scholz, V. Thurner

"Die Praxis der Softwareentwicklung: Eine Erhebung"

Informatik Spektrum 22 (1999) S. 24 - 36

- **[DO96]**

Jörg Desel, Andreas Oberweis:

"Petri-Netze in der Angewandten Informatik

Einführung, Grundlagen und Perspektiven"

Wirtschaftsinformatik 38 (1996) 4, S. 359 - 366

- **[FGH88]**

Fernando Flores, Michael Graves, Brad Hartfield, Terry Winograd

"Computer Systems and the Design of Organizational Interaction"

ACM Transaction on Office Information Systems 6 (1988) 2, S. 153 - 172

- **[FK98]**

Christiane Floyd, Ralf Klischewski:

„Modellierung - ein Handgriff zur Wirklichkeit. Zur sozialen Konstruktion und Wirksamkeit von Informatik-Modellen“

Klaus Pohl, Andy Schürr, G. Vossen, (Hrsg.): Modellierung 98- Proceedings. Universität Münster 1998, S.21-26

- **[FK99]**

Christiane Floyd, Ralf Klischewski

"Informatik - eine Standortbestimmung

Prüfungsunterlagen (Teil1 + Teil2)

Hamburg 1999

- **[FKR97]**
Christiane Floyd, Anita Krabbel, Sabine Ratuski, Ingrid Wetzel:
„Zur Evolution der evolutionären Systementwicklung:
Erfahrungen aus einem Krankenhausprojekt“
Informatik Spektrum, Band 20, Heft 1, Februar 1997, S.13-20
- **[Flo92]**
Christiane Floyd:
„Software Development as Reality Construction“
In: C. Floyd , H. Züllighoven, R. Budde. R. Keil-Slawik (Hrsg.)
"Software Development and Reality Construction"
Springer Verlag, Berlin, Heidelberg, New York, Paris (1992), S. 86 -100
- **[Flo99]**
Christiane, Floyd:
„Software Development Process: Some Refletions on ist Cultural, Political and Ethical As-
pects from a Constructivist Epistemology Point of View“
In: Cybernetics & Human Knowing, Vol.6. No.2., 1999 Seiten: 5-18
- **[Fuc92]**
Klaus Fuchs-Kitowki:
"Theorie der Infromatik im Spannungsfeld zwischen formalem und nicht formaler Welt"
In: Wolfgang Coy et al. (Hrsg.): "Sichtweisen der Informatik", Vieweg, (1992), S. 71 - 83
- **[GKW98]**
Guido Gryczan, Anita Krabbel, Ingrid Wetzel, Heinz Züllighoven:
„ Implizite und explizite Kooperationsmodelle“
In: Th. Herrmann, K. Just-Hahn (Hrsg.): Groupware und organisatorische Innovation, Ta-
gungsband der D-CSCW' 98, B. G. Teubner Stuttgart. Leipzig, 1998
- **[GM97]**
Jonathan Grudin, M. Lynne Markus:
"Organizational Issues in Development and Implementation of Interactive Systems“
In Handbook of Human-Computer Interaction.
M. Helander, T. K. Landauer, P. Prabhu (Hrsg.)
Elsevier, 1997, S. 1457 - 1474
- **[GMD94]**

„Die Revolution in der Unternehmenskultur -
Herausforderung für die Informationstechnik“

In: Der GMD-Spiegel 3 (1994), S. 12-46

- **[Gog94]**

J.A., Goguen:

"Requirements Engineering as the Reconciliation of Technical and Social Issues"

In: M. Jirotko, J.A. Goguen (eds.):

"Requirements Engineering: Social and Technical Issues"

London, Academic Press 1994, S. 165-200

- **[Gra96]**

Grant Harris

"So, What Is This Language Action Approach All About, Anyway?"

<http://www.workframe.com/WFI:Corp>

- **[Gry96]**

Guido Gryczan:

„Prozeßmuster zur Unterstützung kooperativer Tätigkeit“

Deutscher Universitäts-Verlag, Wiesbaden (1996)

- **[HC94]**

M. Hammer, J. Champy:

"Business Process Reengineering"

Campus Verlag, Frankfurt, New York (1994)

- **[Her97]**

Thomas Herrmann:

„Communicable Models for Cooperative Processes“

In: G. Slavedy (ed.). HCI International '97. Proceedings of the 7th International Conference on Human Computer Interaction, San Francisco. Amstersam: Elsevier (1997), S. 185 - 188

- **[Her99]**

Thomas Herrmann:

„Flexible Repräsentation von Prozeßmodellen“

In Proceedings der 9. Fachtagung Software-Ergonomie '99 (1999), S. 123- 136

- **[HHL97]**

Thomas Herrmann , Marcel Hofmann, Kai-Uwe Loser:

„Modellierungsnotationen für prospektive, gestaltungsorientierte Technikfolgenabschätzung“

In Paul Hansjürgen (Hrsg): „Modellierung von Aufbau- und Ablauforganisation:

von der Technozentrik zur Antropozentrik“

Projektbericht des Instituts Arbeit und Technik 97 (1997) S. 33 – 46

- **[HHL99]**
Thomas Herrmann, Marcel Hoffmann, Kai –Uwe Loser:
Modellieren mit SeeMe –
Alternativen zur Trockenlegung feuchter Informationslandschaften“
In: Jörg Desel, Klaus Pohl, Andy Schürr (Hrsg.) "Modellierung '99" (1999) S. 59-74
- **[HHL2000]**
Thomas Herrmann, Marcel Hoffmann; Kai-Uwe Loser , Klaus Moysich:
"Semistructured models are surprisingly useful for user-centered design“
Accepted at Coop2000, Sophia-Antipolis, France, May 2000
"http://www. Informatik.uni-dortmund.de/pubs.html
- **[HL98]**
Thomas Herrmann, Kai-Uwe Loser:
„Vages Modellieren“
Bericht des Fachgebiets Informatik & Gesellschaft, Universität Dortmund. Mai 1998
- **[HSW98]**
T. Herrmann, A.-W. Scheer, Weber H. (Hrsg)
"Verbesserung von Geschäftsprozessen mit flexiblen Workflow-Management-Systemen.
Band 1. Von der Erhebung zum Sollkonzept. Band 2. Von der Sollkonzeptentwicklung zur
Implementierung von Workflow-Management-Anwendungen.
Heidelberg, 1998
- **[IDS99]**
<http://www.ids-scheer.de>
- **[JBS97]**
Stefan Jablonki, Markus Böhm, Wolfgang Schulze (Hrsg.):
„Workflow-Management: Entwicklung von Anwendungen und Systemen:
Facetten einer neuen Technologie“
Heidelberg (1997)
- **[JV87]**
Eike Jessen, Rüdiger Valk:
Rechensysteme: Grundlagen der Modellbildung
Berlin, Heidelberg: Springer (1987)
- **[KKJ98]**

H Kühn, S Junginger, S, D Karagiannis, C Petersen:

„Metamodellierung im Geschäftsprozeßmanagement:

Konzepte, Erfahrungen und Potentiale“

In: Jörg Desel, Klaus Pohl, Andy Schürr (Hrsg.) Modellierung 99, Teubner, 1999, S.75-89

- **[Kli96]**

Ralf Klischewski

"Anarchie - ein Leitbild für die Informatik

Von den Grundlagen der Beherrschbarkeit zur Selbstbestimmten Systementwicklung

Europäische Hochschulschriften: Reihe 41, Informatik Band 24,

Frankfurt am Main: Peter Lang, (1996)

- **[Krc97]**

Helmut Krcmar

"Informationsmanagement"

Springer, Berlin, Heidelberg, New York

- **[KW98]**

Anita Krabbel, Ingrid Wetzel:

„Analyse bereichsübergreifender Prozesse im Rahmen der Auswahl Krankenhausinformationssysteme“

In E., Pinter, E. Swart, K.D. Vitt (Hrsg.): Umfassendes Qualitätsmanagement - Beispiele und Erfahrungsberichte für die Einführung im Krankenhaus (1998), S. 219-233

- **[KW99]**

Ralf Klischewski, Ingrid Wetzel: (1999):

„Cooperation Modeling for Relating System Development and Organizational Change“ Arbeitsbericht, Universität Hamburg, FB Informatik (1999)

- **[KWJ99]**

R. Klischewski, B. Wolff, I. Buhse-Jackewitz, H: Tewis

"Projekt Prüfungsverwaltung (ProPrüVer): "Ergebnisse der Ist-Analyse als Grundlage für die Einführung eines computerunterstützten Systems in der Prüfungsverwaltung der Universität Hamburg" (1999)

Hamburger Informatik Technologie-Center e.V., 1999

- **[Kyn91]**

Morten Kyng:

"Designing for Cooperation: Cooperation in Design"

Communication of the ACM 34 (1991) 12, S.65 - 73

- **[Kyn95]**

Morten Kyng

"Making Representations Work"

Communication of the ACM 38 (1995) 9, S. 46 - 55

- **[LHM95]**

Franz Lehner, Knut Hildebrand, Roland Maier

"Wirtschaftsinformatik: theoretische Grundlagen"

Hanser, München, Wien, 1995

- **[Lex97]**

Hans-Jochen Schneider (Hrsg.)

"Lexikon Informatik und Datenverarbeitung"

4. Auflage, München; Wien: Oldenbourg, 1997

- **[Loo99]**

"Projekt Look"

<http://iundg.cs.uni-dortmund.de/look/Presse/isst.html>

- **[LSW97]**

Peter Langner, Christopf Schneider, Joachim Wehler

"Ereignisgesteuerte Prozeßketten und Petri-Netze"

Berichte des Fachbereichs Informatik, Bericht Nr. 196, Report No. 196 FBI-HH-196/97,
Hamburg

- **[Mar99]**

Declan Martin:

"ActionWorkflow Approach to Process Modelling"

http://www.broadcom.ie/communicative/vol_3_iss_2/papers2/DMN.HTML

- **[Mor92]**

Raul Medina-Mora:

"ActionWorkflow Technology and Applications for Groupware"

In Groupware 92, S.165 -167

- **[MW94]**

Andrea Mosher, Patricia Wall:

"Representations of Work: Bringing Designers and Users Together"

In: R. Trigg, S. I. Anderson, E. A. Dykstra-Erickson (Eds.)

"PDC '94: Proceedings of the Participatory Design Conference."

Chapel Hill, North Carolina, USA

- **[MWR92]**
R. Medina-Mora, T. Winograd, R. Flores, F. Flores:
„The ActionWorkflow Approach to Workflow Management Technology“
In: Conference on CSCW '92, ACM, New York, (1992), S. 281 - 288
- **[NSZ95]**
Markus Nüttgens, August Wilhelm Scheer, Volker Zimmermann:
„Rahmenkonzepte für ein integriertes Geschäftsprozeßmanagement“
Wirtschaftsinformatik 37 (1995) 5, S. 426 – 434

- **[Obe87]**
Horst Oberquelle:
„Sprachkonzepte für benutzergerechte Systeme“
Springer, Berlin (1987)
- **[Obe91]**
Horst Oberquelle:
„Kooperativer Arbeit und Computerunterstützung“
Verlag für Angewandte Psychologie, Göttingen, Stuttgart (1991)
- **[Obe94]**
Horst Oberquelle
"Situationsbedingte und benutzerorientierte Anpaßbarkeit von Groupware"
In Anje Hartmann (Hrsg.)
"Menschengerechte Groupware: Softwareergonomische Gestaltung und partizipativer Umsetzung"
Stuttgart. Teubner 1994, S.
- **[Obe96]**
Andreas Oberweis:
„Modellierung und Ausführung von Workflows mit Petri-Netzen“
Teubner, Stuttgart (1996)
- **[Oes98]**
Bernd Oestereich
"Objektorientierte Geschäftsprozeßmodellierung mit der UML"

In Objektspektrum 2(1998), S. 48 –54

- **[Pie91]**
Ulrich Piepenburg:
„Ein Konzept von Kooperation und seine Implikationen für die technische Unterstützung kooperativer Arbeitsprozesse“
In: H. Oberquelle (Hrsg.) Kooperative Arbeit und Computerunterstützung: Stand und Perspektiven. Stuttgart: Verlag für Angewandte Psychologie, 1991, S. 79 - 98
- **[PM98]**
Hansjürgen Paul, Irene Maucher (Hrsg)
"Inegration von Mensch, Organisation und Technik: eine partzielle Bilanz. Beiträge zum EMISA-Fachgruppentreffen 1998. Graue Reihe des IAT (1998-04). Gelsenkirchen: Institut Arbeit und Technik (1998)
- **[Rei82]**
Wolfgang Reisig:
„Petrietze - Eine Einführung.“
Springer, Berlin (1982)
- **[Rob97]**
Mike Robinson:
„As real as it gets..." Taming models and reconstructing procedures“
In Social science, technical systems, and cooperative work: beyond the great divide. Bowker, G. (ed.) Lawrence Erlbaum Associates, Mahwah NJ (1997), S. 257 - 274
- **[Rol98]**
Arno Rolf:
"Grundlagen der Organisations- und Wirtschaftsinformatik"
Springer Berlin (1998)
- **[Sac95]**
Patricia Sachs
"Transforming Work: Collaboration, Learning and Design"
Communication of the ACM 38 (1995) 9, S. 36 - 44
- **[Sch96]**
August-Wilhelm Scheer:
„ARIS-Toolset: Von Forschungs-Prototypen zum Produkt“
Informatik-Spektrum 19 (1996), S. 71 - 78
- **[Sch98a]**

August-Wilhelm Scheer:

„ARIS- Vom Geschäftsprozeß zum Anwendungssystem“

Springer, Berlin (1998)

- **[Sch98b]**

August-Wilhelm Scheer:

„ARIS- Modellierungsmethoden, Metamodelle., Anwendungen“

Springer, Berlin (1998)

- **[Suc95]**

Lucy Suchman : (Hrsg.)

„Representations of Work“

Communication of ACM 38 (1995) 9

- **[Suc95a]**

"Making Work Visible"

Communication of ACM 38 (1995) 9, S. 56 - 63

- **[Sza98]**

Constantin Szallies:

"Software-Spezifikation mit ereignisgesteuerten Prozeßketten (EPK)

Softwaretechnik - Trends 18 (1998) 2, S. 38 -43

- **[Val97]**

Rüdiger Valk:

„ Die Informatik zwischen Formal- und Humanwissenschaften.

Informatik Spektrum 20, (1997) S. 95 – 100

- **[Wah98]**

"UML kompakt"

In OBJEKTspektrum 2(1998), S. 22 - 33

- **[WF86]**

Terry Winograd, Fernando Flores:

"Understanding Computers and Cognition. A New Foundation for Design"

Norwoos (1986)

- **[Win87]**

Terry Winograd:

„A Language/Action Perspektive on the Design of Cooperative Work“

In: Human- Computer Interaction, 3 (1987, S. 3 - 30

- **[WF89]**
Terry Winograd, Fernando Flores:
„Erkenntnis Maschinen Verstehen. Zur Neugestaltung von Computersystemen“
Rotbuch, Berlin (1989)
- **[WKK99]**
B. Wolff, K. Fuchs-Kittowski, R. Klischewski, A. Möller, A. Rolf
"Organisationstheorie als Fenster zur Wirklichkeit"
In J. Becker, W.König., R.Schütte, O. Wendt, S. Zelewski (Hrsg)
"Wirtschaftsinformatik und Wissenschaftstheorie. Bestandsaufnahme und Perspektiven"
Wiesbaden 1999, S. 291 – 330
- **[WKKL98]**
"Kooperation für Software für Kooperation
Erfahrungen aus einem partizipativen Softwareprojekt"
In Claus, V. (Hrsg) Informatik und Ausbildung.
GI-Fachtagung 98 Informatik und Ausbildung, Stuttgart 30. März bis 1. April 1998,
Springer Verlag (Reihe Informatik aktuell), S. 73 - 81, (1998)
- **[Zü198]**
Heinz Züllighoven:
„Das objektorientierte Konstruktionshandbuch nach dem Werkzeug & Material – Ansatz“
Dpunkt-Verlag (1998)

Hiermit erkläre ich, daß die vorliegende Diplomarbeit von mir selbständig erstellt wurde und daß ich nur die angegebenen Quellen und Hilfsmittel verwendet habe.

Nilgün Özek
Matrikel-Nr: 4175480
Alter Zirkusplatz 15
22880 Wedel

Wedel, 08.06.2000

