

Diplomarbeit

*Komponenten zur Kooperationsunterstützung:  
Das Postversandsystem*

Mirko Freund  
Vor der Mühle 12  
21640 Horneburg

Juli 2000

Erstbetreuung: Dr. Ingrid Wetzel  
Zweitbetreuung: Prof. Dr. Horst Oberquelle

Fachbereich Informatik  
Universität Hamburg  
Vogt-Kölln-Straße 30  
22527 Hamburg

**Erklärung:**

Ich versichere hiermit, diese Arbeit selbständig und unter ausschließlicher Zuhilfenahme der in der Arbeit aufgeführten Hilfsmittel erstellt zu haben.

Hamburg, den

**Mirko Freund**

Vor der Mühle 12  
21640 Horneburg

**Betreuung:**

Dr. Ingrid Wetzel (Erstbetreuerin)  
Prof. Dr. Horst Oberquelle (Zweitbetreuer)

**Dr. Ingrid Wetzel**

Arbeitsbereich Softwaretechnik  
Fachbereich Informatik  
Universität Hamburg  
Vogt-Kölln-Str. 30  
22175 Hamburg

**Prof. Dr. Horst Oberquelle**

Arbeitsbereich Angewandte und Sozialorientierte Informatik  
Fachbereich Informatik  
Universität Hamburg  
Vogt-Kölln-Str. 30  
22175 Hamburg

# Inhaltsverzeichnis

<b>1</b>	<b>EINLEITUNG .....</b>	<b>1</b>
1.1	COJAC-PROJEKT .....	1
1.2	ZIELSETZUNG .....	2
1.3	AUFBAU DER ARBEIT .....	3
1.4	GRAPHISCHE UND TEXTUELLE KONVENTIONEN .....	4
1.5	DANKSAGUNG .....	5
<b>2</b>	<b>ANSÄTZE ZUR UNTERSTÜTZUNG KOOPERATIVER ARBEIT .....</b>	<b>7</b>
2.1	KOOPERATIVE ARBEIT .....	7
2.2	UNTERSTÜTZUNGSARTEN.....	9
2.2.1	<i>Kommunikationsunterstützung</i> .....	10
2.2.2	<i>Koordinationsunterstützung</i> .....	11
2.2.3	<i>Kooperationsunterstützung</i> .....	13
2.3	TIME-SPACE-MATRIX .....	17
2.4	ZUSAMMENFASSUNG UND ZIEL .....	19
<b>3</b>	<b>WAM-ANSATZ.....</b>	<b>23</b>
3.1	WAM-ANSATZ ALS SOFTWAREENTWICKLUNGSMETHODE .....	23
3.2	LEITBILDER .....	25
3.3	ENTWURFSMETAPHERN .....	26
<b>4</b>	<b>ANFORDERUNGEN AN DAS POSTVERSANDSYSTEM .....</b>	<b>31</b>
4.1	FACHLICHE ANFORDERUNGEN .....	31
4.2	TECHNISCHE ANFORDERUNGEN .....	34
4.3	BENUTZUNGSMODELLE .....	35
4.3.1	<i>Einfacher Umgang</i> .....	35
4.3.2	<i>Erweiterter Umgang</i> .....	37
4.3.3	<i>Konfiguration</i> .....	38
4.4	ARCHITEKTUREIGENSCHAFTEN .....	40
4.4.1	<i>Rahmenwerk</i> .....	40
4.4.2	<i>Erweiterbarkeit</i> .....	41
4.4.3	<i>Anpaßbarkeit</i> .....	42
4.4.4	<i>Wiederverwendbarkeit</i> .....	44
4.4.5	<i>Kombinierbarkeit</i> .....	44
<b>5</b>	<b>KONZEPTION DES POSTVERSANDSYSTEMS.....</b>	<b>47</b>
5.1	ENTWURF .....	47
5.2	FRAGESTELLUNGEN BEI DER KONZEPTION .....	53
5.2.1	<i>Adressierung</i> .....	53
5.2.2	<i>Transportierbarkeit von Materialien</i> .....	55
5.3	KOMPONENTEN .....	63
5.3.1	<i>Handhabung</i> .....	66
5.3.1.1	Posteingangsautomat.....	68
5.3.1.2	Posteingangsstrategien .....	70
5.3.1.3	Postausgangsautomat .....	71
5.3.1.4	Postein- und Postausgangskorb .....	72
5.3.1.5	Transportbehälter .....	73
5.3.1.6	Exkurs: Rohrpost .....	74
5.3.2	<i>Postzustelldienst</i> .....	75
5.3.2.1	Postzustelldienst-Client.....	78
5.3.2.2	Postzustelldienst-Server .....	79
5.3.3	<i>Postdienststelle</i> .....	80
5.3.3.1	Postfachwand .....	83
5.3.3.2	Postbedienstete.....	84
5.4	WERKZEUGUNTERSTÜTZUNG .....	85
5.4.1	<i>Adreßeditor</i> .....	87

5.4.2	<i>Posteingangskorbeditor</i> .....	87
5.4.3	<i>Postausgangskorbeditor</i> .....	88
5.4.4	<i>Einstellwerkzeug des Posteingangsautomaten</i> .....	89
<b>6</b>	<b>INTEGRATION INS JWAM-RAHMENWERK</b> .....	<b>93</b>
6.1	WAM-MODELLARCHITEKTUR .....	93
6.2	ARCHITEKTUR DES JWAM-RAHMENWERKES UND EINORDNUNG DES POSTVERSANDSYSTEMS .....	98
<b>7</b>	<b>ZUSAMMENFASSUNG UND AUSBLICK</b> .....	<b>103</b>
	<b>ABBILDUNGSVERZEICHNIS</b> .....	<b>109</b>
	<b>LITERATURLISTE</b> .....	<b>111</b>

# 1 Einleitung

Aufgrund der zunehmenden Verwendung von Computersystemen in Wirtschaft und Wissenschaft ist eine angemessene Unterstützung kooperativer Arbeitsprozesse ein aktuelles Forschungs- und Anwendungsfeld der Informatik. Es ist immer wichtiger, nicht nur die Aufgabenerledigung am Einzelarbeitsplatz zu unterstützen, sondern auch die notwendige Kooperation.

Die vorliegende Arbeit ist von mir am Arbeitsbereich Softwaretechnik des Fachbereichs Informatik der Universität Hamburg erstellt worden. An diesem Arbeitsbereich wird der Softwareentwicklungsansatz WAM vertreten, der auch Kontext dieser Arbeit ist. Der WAM-Ansatz ist eine Methode, die eine objektorientierte evolutionäre Herangehensweise bei der Entwicklung interaktiver Anwendungssysteme vertritt. Eine besondere Eigenschaft dieser Methode ist, daß sie ein Leitbild mit Entwurfsmetaphern verwendet. Das zugrundeliegende Leitbild ist der Arbeitsplatz für eigenverantwortliche Expertentätigkeit, das mit den Metaphern Werkzeug, Automat und Material ausgestaltet wird.

Auch bei der Entwicklung des WAM-Ansatzes stand ursprünglich der Einzelarbeitsplatz im Mittelpunkt, d.h. die Unterstützung der individuellen Arbeit. Es wurde jedoch erkannt, daß nur ein Teil der von einem Anwendungsexperten zu verrichtenden Tätigkeiten individuelle Arbeit ist. Aus diesem Grund ist später dieser Ansatz um die Unterstützung der kooperativen Arbeit erweitert worden (vgl. [Wul95], [Gry96], [Zül98]).

Um die Unterstützung kooperativer Arbeit im Kontext des WAM-Ansatzes weiter auszubauen, wurde das COJAC-Projekt initiiert. Im Rahmen dieses Projektes wurden drei Ansätze zur Kooperationsunterstützung aufgegriffen, konzipiert und realisiert. Die vorliegende Arbeit widmet sich einem dieser Ansätze.

## 1.1 COJAC-Projekt

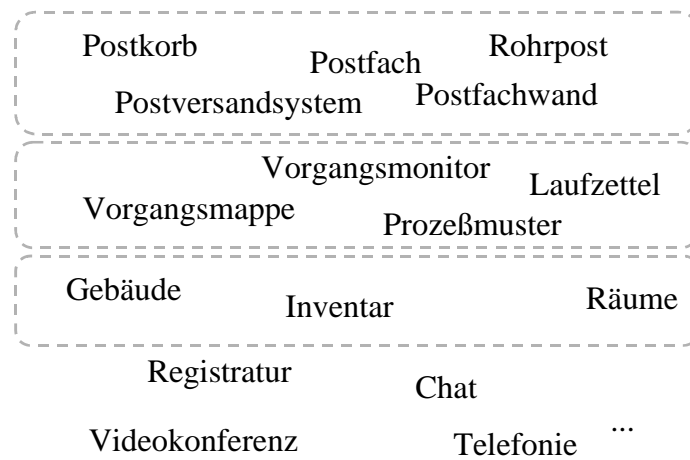
Das COJAC-Projekt begann im März 1999. Dabei steht der Name COJAC für die Initialen von "Cooperative JWAM Architecture Components". Dieser Titel steht in einem engen Zusammenhang mit dem Ziel des Projektes, auf das ich im folgenden noch eingehen werde.

Der Ausgangspunkt des Projektes war, daß das JWAM-Rahmenwerk, das der Entwicklung interaktiver Anwendungssysteme nach dem WAM-Ansatz in der Programmiersprache JAVA dient, kooperative Arbeit nicht umfassend unterstützt. Es gab bisher nur zwei Arbeiten, die in diese Richtung gingen. So wurde die explizite Kooperation mittels Materialaustausch durch ein einfaches Transportsystem (vgl. [Fre98]) unterstützt, während für die implizite Kooperation eine Registratur (vgl. [Hav99]) diente.

Aus diesem Grund wurde als Ziel für das Projekt bestimmt, Komponenten für das JWAM-Rahmenwerk zu entwickeln, welche die kooperative Arbeit unterstützen. Dabei sollen die Komponenten so gestaltet werden, daß sie als Basis in verschiedenen Anwendungskontexten eingesetzt werden können.

Zu Beginn des Projektes wurden eine Vielzahl von verschiedenen Unterstützungsmöglichkeiten in Form von Kooperationsmitteln und -medien

zusammengetragen, um zu entscheiden, welche davon realisiert werden sollten (siehe Abbildung 1).



**Abbildung 1: Auswahl verschiedener Kooperationsmittel und -medien des COJAC-Projektes**

Wie in der Abbildung zu erkennen ist, lassen sich die verschiedenen Kooperationsmittel und -medien gruppieren, da sie in einem engen Zusammenhang stehen. So haben wir uns für die drei in der Abbildung markierten Bereiche entschieden, die stellvertretend für verschiedene Themengebiete stehen. Jeder der Projektmitglieder hat sich im Rahmen seiner Diplomarbeit einem dieser Themen gewidmet. So hat Jörn Koch die Konzeption und Realisierung eines Raumsystems aufgegriffen (vgl. [Koc00]), während Holger Breitling sich der Koordinationsunterstützung routinierter arbeitsteiliger Prozesse angenommen hat (vgl. [Bre00]). Die vorliegende Arbeit widmet sich dem in der Abbildung obersten eingerahmten Bereich, indem eine Kooperationsunterstützung mittels Materialaustausch in Form eines Postversandsystems konzipiert und realisiert wird. Andere Unterstützungsmöglichkeiten haben wir im Rahmen des Projektes nicht weiter verfolgt.

Die Auswahl der Themengebiete mag auf den ersten Blick willkürlich erscheinen, jedoch besteht zwischen ihnen ein Zusammenhang. So sind z.B. Vorgangsmappen mit Laufzetteln erst dann sinnvoll einsetzbar, wenn sie über ein Postversandsystem versendet werden können. Weiterhin wäre auch ein Vorgangsmonitor überflüssig, wenn Vorgänge repräsentiert durch Vorgangsmappen nicht arbeitsplatzübergreifend versendet werden könnten. Auch der oberste und unterste markierte Bereich in der Abbildung ergänzen sich. So können z.B. Räume mit Postkörben ausgestattet werden. Damit wäre es u.a. möglich, Materialien zwischen verschiedenen privaten Räumen auszutauschen.

## 1.2 Zielsetzung

Ziel dieser Arbeit ist es, einen softwaretechnischen Beitrag zur Unterstützung kooperativer Arbeit zu leisten. Dieser Beitrag besteht aus der Konzeption und Realisierung eines Softwaresystems, das als Basisunterstützung in verschiedenen Anwendungskontexten wiederverwendet werden kann.

Aus konzeptioneller Sicht soll erarbeitet werden, wie im Rahmen des WAM-Ansatzes eine Unterstützung kooperativer Arbeit gestaltet werden kann. Dabei sollen Anforderungen aufgestellt werden, die eine Differenzierung zu vergleichbarer Groupware, wie z.B. einer herkömmlichen elektronischen Post<sup>1</sup>, erlauben. Neben den Anforderungen muß auch eine Vorstellung von der Handhabung und Präsentation des Softwaresystems entwickelt werden, die an verschiedene Anwendungskontexte adaptierbar ist.

Damit das Softwaresystem als eine Basisunterstützung für kooperative Arbeit in verschiedenen Anwendungskontexten eingesetzt werden kann, muß dessen Softwarearchitektur entsprechende Eigenschaften aufweisen. Diese Eigenschaften sind zu evaluieren. Dabei sollen Ansätze aufgezeigt werden, wie diese Eigenschaften Ausdruck in der Softwarearchitektur finden können. Weiterhin besteht der Anspruch, daß das Softwaresystem in das JWAM-Rahmenwerk integriert werden soll.

Im Rahmen des COJAC-Projektes habe ich zur Realisierung einer Unterstützung kooperativer Arbeit das Ziel, ein Postversandsystem zu entwickeln. Dieses soll Kooperationsmittel und -medien sowie eine Infrastruktur aus Werkzeugen, Automaten und Materialien anbieten, mit denen der Arbeitsplatz des Benutzers ausgestaltet werden kann und die ihm bei der eigenverantwortlichen Aufgabenerledigung kooperativer Arbeitsprozesse unterstützen. Weiterhin soll vorgestellt werden, wie die dargelegten Softwarearchitektureigenschaften für eine Basisunterstützung beim Postversandsystem umgesetzt wurden.

All die genannten Zielvorgaben möchte ich wie folgt zusammenfassen.

Das Ziel dieser Arbeit ist die Konzeption und die Realisierung eines Postversandsystems im Kontext des WAM-Ansatzes zur **Unterstützung kooperativer Arbeit mittels Materialaustausch** und dessen **Integration ins JWAM-Rahmenwerk**.

Nachdem ich in Kapitel 2 weitere Unterstützungsmöglichkeiten kooperativer Arbeit aufgezeigt und klassifiziert habe, werde ich in Abschnitt 2.4 dieses Ziel erneut aufgreifen und konkretisieren.

### **1.3 Aufbau der Arbeit**

Um ein Verständnis für das Ziel dieser Arbeit zu schaffen, befaßt sich Kapitel 2 mit verschiedenen Ansätzen zur Kooperationsunterstützung und beginnt mit einer Definition des Begriffes "kooperative Arbeit". Hierbei wird deutlich, welche Möglichkeiten der Unterstützung es gibt. Um meinen Ansatz einordnen zu können, zeige ich zwei Klassifizierungen für kooperationsunterstützende Software auf. Anschließend fasse ich zusammen und konkretisiere mein Ziel, ein Postversandsystem zur Unterstützung asynchroner verteilter Kooperation mittels Materialaustausch zu schaffen.

Der softwaretechnische Hintergrund dieser Arbeit ist der Softwareentwicklungsansatz WAM, der meine Herangehensweise bei der Entwicklung einer Kooperationsunterstützung geprägt hat. Aus diesem Grund stellt Kapitel 3 diesen Ansatz vor und erläutert, wie er eine evolutionäre Vorgehensweise, anwendungsorientierte Dokumente, Leitbilder mit Entwurfsmetaphern sowie Konstruktionstechniken vereinigt, um eine hohe Akzeptanz des Anwendungssystems bei den Benutzern zu erreichen. Hierbei hebe ich das Leitbild vom

---

<sup>1</sup> Eine geläufige Bezeichnung für die elektronische Post ist "E-Mail".

Arbeitsplatz für eigenverantwortliche Expertentätigkeit hervor und beschreibe, wie es mit den Entwurfsmetaphern Werkzeug, Automat, Material und Umgebung ausgestaltet wird.

Aufgrund meiner Zielsetzung und dem Kontext des WAM-Ansatzes stelle ich in Kapitel 4 die Anforderungen an das Postversandsystem auf. Dabei wird zwischen fachlichen und technischen Anforderungen unterschieden.

Die fachlichen Anforderungen beziehen sich darauf, was das Postversandsystem gegenüber dem Benutzer leisten soll, wie es sich ihm präsentiert und welche Vorteile es gegenüber einer herkömmlichen elektronischen Post aufweist. Weiterhin skizziere ich drei Benutzungsmodelle, die das Postversandsystem unterstützen soll.

Die technischen Anforderungen beziehen sich auf die Bedürfnisse der Softwareentwickler, die das Postversandsystem wiederverwenden, erweitern bzw. an den jeweiligen Anwendungskontext anpassen wollen. Ich diskutiere Eigenschaften, die die Softwarearchitektur des Postversandsystems aufweisen muß, damit es als Basisfunktionalität für die explizite Kooperation mittels Materialaustausch eingesetzt werden kann.

Ausgehend von den Anforderungen konzipiere ich in Kapitel 5 das Postversandsystem. Dabei werden zuerst der Entwurf hergeleitet und anschließend im einzelnen dessen Komponenten erläutert. Insbesondere gehe ich hierbei auf die Schnittstellen der Komponenten ein und beschreibe deren internen Aufbau. Abschließend wird in Kapitel 5 der Umgang mit den einzelnen Werkzeugen erklärt, die Bestandteile des Postversandsystems sind, da sie größtenteils die geforderten Benutzungsmodelle verkörpern.

Das JWAM-Rahmenwerk unterstützt die Anwendungsentwickler bei der Erstellung von interaktiven Anwendungssystemen nach dem WAM-Ansatz. Eine der technischen Anforderungen ist, daß das Postversandsystem in das JWAM-Rahmenwerk integriert wird. Kapitel 6 beginnt aus diesem Grund mit der Beschreibung der WAM-Modellarchitektur, welche die Vorstellung über den Aufbau von Rahmenwerken prägt. Anschließend wird gezeigt, wie der Aufbau des JWAM-Rahmenwerks sich an der WAM-Modellarchitektur orientiert. Dabei gehe ich auf die Schichtenarchitektur des JWAM-Rahmenwerkes ein und nenne ausgewählte Sub-Rahmenwerke der jeweiligen Schichten. Mit diesem Verständnis über den Aufbau des JWAM-Rahmenwerkes lege ich dar, in welcher Schicht das Postversandsystem integriert wurde.

Die Ergebnisse dieser Arbeit werden in Kapitel 7 zusammengefaßt. Dabei gehe ich auf den Zusammenhang des Postversandsystem mit den anderen Systemen des COJAC-Projektes ein. Weiterhin wird aufgezeigt, in welchen Punkten das Postversandsystem ausgestaltet bzw. weiterentwickelt werden kann.

#### **1.4 Graphische und textuelle Konventionen**

Die technisch konstruktiven Teile der vorliegenden Arbeit basieren auf den Konzepten der objektorientierten Anwendungsentwicklung nach dem WAM-Ansatz. Somit entstammen auch die verwendeten Begriffen diesen Konzepten. In diesem Zusammenhang wird bei der Beschreibung von Entwurfskomponenten u.a. von Klassen, Ober- und Unterklassen, Aspekten, Schnittstellen und von Vererbung gesprochen.

Für die Darstellung der Entwurfskomponenten in Form von Klassendiagrammen wird in dieser Arbeit die UML-Notation (vgl. [BRJ99]) verwendet. In einigen Klassendiagrammen



habe ich davon abweichend die Klassen, die als Eingangsschnittstellen einer Komponente dienen, fett und schwarz umrandet, während die Ausgangsschnittstellen zusätzlich grau hinterlegt sind (siehe Abbildung 2).



**Abbildung 2: Grafische Hervorhebung von Ein- und Ausgangsschnittstellen einer Komponente**

Für die Namen von Entwurfskomponenten habe ich generell die *Courier*-Schrift gewählt. Somit bezeichnen alle Begriffe, die in dieser Schriftart dargestellt sind, Klassen meines konstruktiv technischen Entwurfs.

Die Arbeit ist in der Übergangszeit zur Neuregelung der deutschen Rechtschreibung entstanden. Aus diesem Grund möchte ich darauf hinweisen, daß die neuen Regeln für diese Arbeit nicht angewendet worden.

## **1.5 Danksagung**

An dieser Stelle soll all denen gedankt werden, die bei der Erstellung dieser Arbeit auf die eine oder andere Art und Weise behilflich waren. Dr. Ingrid Wetzels danke ich für die Erstbetreuung der Arbeit und die vielen Anregungen und Hinweise. Bei Prof. Dr. Horst Oberquelle bedanke ich mich für die Zweitbetreuung.

Für die Unterstützung, Anregungen und Diskussionen beim Entwurf und der Realisierung des Postversandsystems bedanke ich mich bei den Mitgliedern des COJAC-Projektes Jörn Koch und Holger Breitling. Bei Henning Wolf und Martin Lippert bedanke ich mich für die Hilfe bei der Integration des Postversandsystems in das JWAM-Rahmenwerk. Die damit verbundenen Reviews haben beigetragen, das Postversandsystem weiterzuentwickeln und Schwachstellen der Implementation zu beseitigen.

Für die unermüdliche Unterstützung bei der Korrektur der vorliegenden Arbeit möchte ich mich bei Eileen Bender und Christine Freund bedanken.



## **2 Ansätze zur Unterstützung kooperativer Arbeit**

Die Unterstützung der kooperativen Arbeit gewinnt zunehmend an Interesse sowohl in der Wirtschaft als auch in der Forschung und Entwicklung. Die Aktivitäten, die in diese Richtung gehen, werden dem CSCW-Bereich zugeordnet. Hierbei steht CSCW für Computer-Supported Cooperative Work. Der Begriff wurde 1984 von I. Greif und P. Cashman beim ersten Workshop zu "Computer-Supported Cooperative Work (CSCW)" am MIT in Cambridge eingeführt. Diese Forschungsrichtung hat ihre Anfänge in den achtziger Jahren gefunden und wurde vor allem in den USA stark vorangetrieben (vgl. [CSCW86], [CSCW88], [CSCW90]). In den neunziger Jahren begann man sich auch in Europa verstärkt für dieses Gebiet zu interessieren, weshalb entsprechende E-CSCW- und D-CSCW-Konferenzen stattfanden (vgl. [ECSCW89], [ECSCW91], [ECSCW93]).

Dieses Kapitel betrachtet verschiedene Klassifikationen von kooperationsunterstützender Software, um das Postversandssystem in der Fülle der vorhandenen Ansätze aus dem CSCW-Bereich einordnen zu können. Zu diesem Zweck unterteilt sich dieses Kapitel in die folgenden Abschnitte:

Abschnitt 2.1 erarbeitet eine Definition kooperativer Arbeit, die Gegenstand der Unterstützung ist. Dabei verdeutliche ich, welche verschiedenen Tätigkeiten zur Kooperation notwendig sind und unterstützt werden können.

Abschnitt 2.2 widmet sich einer Strukturierung von Unterstützungsmöglichkeiten in Anlehnung an den zuvor thematisierten Begriff der kooperativen Arbeit. Ich gehe insbesondere auf die einzelnen Schichten (Kooperations-, Koordinations- und Kommunikationsunterstützung) detailliert ein. Diese Struktur dient u.a. Groupware zu klassifizieren und zeigt somit ein Spektrum für Unterstützungsmöglichkeiten auf.

Abschnitt 2.3 behandelt eine weitere weit verbreitete Klassifizierung von Groupware, welche die Strukturierung von Unterstützungsmöglichkeiten sinnvoll ergänzt. In dieser Klassifizierung stehen nicht die zur Kooperation notwendigen Tätigkeiten im Vordergrund, sondern die dabei möglichen zeitlichen sowie örtlichen Rahmenbedingungen.

Abschnitt 2.4 verdeutlicht auf Grundlage der vorangegangenen Erörterungen, in welchem Rahmen eine Unterstützung kooperativer Arbeit durch ein Postversandssystem geschaffen werden soll. Mit diesen Erkenntnissen konkretisiere ich das Ziel, das für den weiteren Verlauf dieser Arbeit maßgeblich ist.

### **2.1 Kooperative Arbeit**

Die Arbeit, die von einem Anwendungsexperten zu erledigen ist, läßt sich in individuell und in kooperativ zu bearbeitende Anteile aufteilen. Um die kooperativen Anteile konkreter fassen zu können, sollen im folgenden unter kooperativer Arbeit:

"[...] Arbeitssituationen verstanden werden, in denen mehrere Personen zusammenarbeiten zwecks Erreichung eines Ergebnisses, welches unter den gegebenen Randbedingungen nur gemeinsam, aber nicht einzeln erzielt werden kann." ([Obe91], S. 4)

Weiterhin lassen sich Eigenschaften wie folgt definieren, die diese Arbeitssituationen charakterisieren:

- " - 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 Aufrechterhaltung eines gegenseitigen Verständnisses und flexibler Anpassung." ([Obe91], S. 4)

Hierbei sind für mich folgende Passagen von Interesse, da sie auf verschiedene Unterstützungsmöglichkeiten kooperativer Arbeit deuten:

1. "[...] in denen mehrere Personen zusammenarbeiten [...]"

Hier wird deutlich, daß wir es nicht mehr nur mit Einzelarbeitsplätzen zu tun haben. Eine mögliche Unterstützung könnte darin bestehen, eine Plattform zur Verfügung zu stellen, auf Grundlage derer die Personen arbeitsteilig ihre Aufgaben erledigen können. Jedoch zeichnet sich hier das Problem ab, daß die Arbeitsplätze verschiedenartig sein können, d.h. eine Computerunterstützung muß einer heterogenen Systemlandschaft entsprechen. Hierbei verstehe ich unter einer heterogenen Systemlandschaft, daß sowohl die verwendete Computerhardware als auch die Software von Arbeitsplatz zu Arbeitsplatz verschieden sein können. Eine adäquate Computerunterstützung kooperativer Arbeit sollte sich in einem solchen Fall nicht auf bestimmte Systemgegebenheiten beschränken.

2. "[...] gemeinsame Nutzung knapper Ressourcen durch Austausch oder gleichzeitige Nutzung [...]"

Bei dieser Passage werden gleich zwei Möglichkeiten für eine Unterstützung offensichtlich. Zum einen sollte der Austausch von Ressourcen ermöglicht werden, d.h. der Transport von einem Ort zu einem anderen, als auch deren synchroner Zugriff. Im weiteren Verlauf dieser Arbeit wird insbesondere der Transport im Vordergrund stehen.

3. "[...] Koordination der Einzelhandlungen [...]"

Hiermit läßt sich festhalten, daß kooperative Arbeit auch Koordination benötigt. Somit dient eine Unterstützung koordinierender Tätigkeiten, wie das Aufstellen von Zeit- oder Ablaufplänen, der Kooperation.

#### 4. "[...] Verständigung [...]"

Darin kommt zum Ausdruck, daß sich die kooperierenden Personen verständigen müssen, d.h. sie müssen miteinander kommunizieren. Das läßt darauf schließen, daß eine Kommunikationsunterstützung der kooperativen Arbeit dienlich ist.

Die herausgestellten Unterstützungsmöglichkeiten für die kooperative Arbeit lassen zwei Dimensionen erkennen. Zum einen gibt es verschiedene Formen der Unterstützung, die eine Struktur bilden (siehe Abschnitt 2.2), und zum anderen verschiedene zeitliche sowie örtliche Rahmenbedingungen (siehe Abschnitt 2.3). Diese zwei Dimensionen erlauben eine Klassifizierung von Groupware. Hierbei wird Groupware von Oberquelle verstanden als:

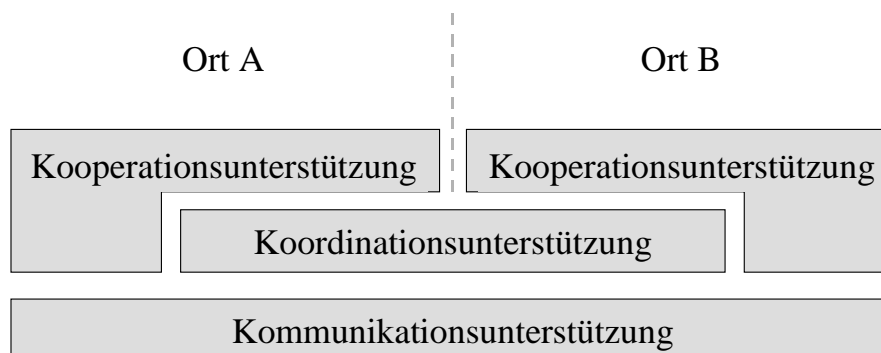
"Mehrbenutzer-Software, die zur Unterstützung von kooperativer Arbeit entworfen und genutzt wird und die es erlaubt,

- Informationen und sonstige Materialien zwischen privaten Räumen ihrer Benutzer koordiniert auszutauschen oder
- gemeinsame Materialien in gemeinsamen Räumen koordiniert zu bearbeiten." ([Obe99], S. 15)

Es sei hier noch angemerkt, daß es eine Vielzahl weiterer Dimensionen zur Klassifizierung von Groupware gibt (vgl. [EGR91]), von denen ich aber im Rahmen dieser Arbeit absehe.

## 2.2 Unterstützungsarten

Wie im vorangegangenen Abschnitt dargestellt wurde, läßt sich Groupware u.a. anhand der Art der Unterstützung unterteilen (vgl. [BBK+99]). So wurden bereits folgende Arten aufgezeigt: Kommunikations-, Koordinations- und Kooperationsunterstützung. Diese einzelnen Unterstützungsformen stehen in Abhängigkeit zueinander und bauen einander auf (siehe Abbildung 3).



**Abbildung 3: Struktur der Unterstützung ([Bur97], S. 39)**

So stellt die Kommunikationsunterstützung die Basis dar, auf der die anderen beiden Unterstützungsarten aufsetzen. Ihr Aktionsbereich ist ortsübergreifend. Die Koordinationsunterstützung basiert lediglich auf der Kommunikationsunterstützung und

agiert ebenfalls ortsübergreifend, während die Kooperationsunterstützung sich der beiden anderen Unterstützungsarten bedient. Die konkreten Unterstützungsarten werden in den folgenden Abschnitten näher erläutert.

### 2.2.1 Kommunikationsunterstützung

Bei der arbeitsteiligen Aufgabenerledigung spielt die zwischenmenschliche Kommunikation eine tragende Rolle. Ohne eine Kommunikation wären die Beteiligten nicht in der Lage, sich zu koordinieren und erst recht nicht miteinander zu kooperieren.

**Kommunikation:**

"Kommunikation ist der Austausch von Informationen." ([Dud93], S. 339)

Hierbei kann die Kommunikation unter verschiedenen Aspekten betrachtet werden. So ist zwischen verbaler und nonverbaler Kommunikation zu unterscheiden. Unter verbaler Kommunikation wird verstanden, wenn mit Worten Informationen ausgetauscht werden. Alle übrigen Formen zählen zu der nonverbalen Kommunikation, wie Gestik, Bilder, Geräusche etc.

Weiterhin spielt der Aspekt Kommunikationsrichtung eine große Rolle. Sie kann uni-, bi- oder multidirektional sein. Dabei bedeutet unidirektional, daß die Kommunikation nur in eine Richtung vonstatten geht. Das ist z.B. beim Radio hören der Fall. Man spricht von einer bidirektionalen Kommunikation, wenn der Informationsaustausch in genau zwei Richtungen geschieht, wie man es bei einem Zwiegespräch vorfindet. Eine multidirektionale Kommunikation liegt vor, wenn es mehrere Kommunikationsrichtungen gibt. Das ist z.B. bei einem Meeting mit mehreren beteiligten Personen der Fall.

Ein weiterer Aspekt der Kommunikation ist die Synchronität. So wird zwischen synchroner und asynchroner Kommunikation unterschieden. Hierbei steht synchron für zeitgleiches Senden und Empfangen von Informationen, wie es bei einem Telefonat der Fall ist. Asynchrone Kommunikation bedeutet, daß das Senden und Empfangen zeitverschieden erfolgt.

Auch bei der Adressierung der Informationen gibt es verschiedene Möglichkeiten. So kann sich die Information direkt an eine Person oder Personengruppe wenden. In diesem Fall spricht man von einer direkten Adressierung. Wenn statt dessen die Information keinen konkreten Empfänger oder Empfängergruppe hat und somit keine Annahmen getroffen werden, wer die Information erhält, handelt es sich um eine anonyme Adressierung.

Bei der Unterstützung der Kommunikation steht der informationsorientierte Nutzen im Vordergrund, der sich auf den Austausch von Informationen beschränkt. Es geht hierbei nicht um die Bearbeitung im Sinne von Veränderung und Auswertung. Allerdings muß gewährleistet sein, daß die Informationen in Form von Daten beliebigen Formates ausgetauscht werden können, um sowohl die verbale als auch die nonverbale Kommunikation zu unterstützen. Beispielhaft für die Unterstützung verbaler Kommunikation ist, wenn das ansonsten gesprochene Wort in Form von geschriebenem Text ausgetauscht wird. Die nonverbale Kommunikation ließe sich u.a. durch Austausch von Bilddaten unterstützen.

Durch die Ausrichtung der Kommunikationsunterstützung auf den rein informationsorientierten Nutzen kann sie sowohl eigenständig als auch im Zusammenspiel

mit weiterer Groupware verwendet werden. An den folgenden Beispielen für Realisierungen der Kommunikationsunterstützung lassen sich die aufgezeigten Eigenschaften der Kommunikation erörtern.

Die elektronische Post dient zum Übermitteln von Textnachrichten. An diesen Nachrichten können zusätzlich noch Dateien angefügt werden, die beliebigen Inhaltes sein können. Damit wird die elektronische Post der Unterstützung der verbalen als auch der nonverbalen Kommunikation gerecht. Die Kommunikationsrichtung ist multidirektional, da eine Nachricht an mehrere Empfänger gleichzeitig gesendet werden kann. Außerdem ist die elektronische Post immer asynchroner Natur, da erst nachdem eine Nachricht abgesendet wurde, sie zu einem späteren undefinierten Zeitpunkt beim Empfänger eintrifft. Alle Nachrichten, die versendet werden sollen, müssen für einen bestimmten Empfänger adressiert werden. Daher ist die direkte Adressierung eine typische Eigenschaft für eine elektronische Post.

Ein Chat-Werkzeug ist eine Computerunterstützung eines Gespräches zwischen mehreren Personen, die sich an verschiedenen Orten befinden. Die Kommunikationsrichtung ist somit multidirektional. Um ein Gespräch zu initiieren, müssen sich die Teilnehmer unter Bekanntgabe ihres Namens in einem virtuellen Chat-Raum einfinden. Das Chat-Werkzeug erlaubt die Eingabe von Text, welcher nahezu zeitgleich auf den Bildschirmen aller Gesprächsteilnehmer zu sehen ist. Daran läßt sich auch erkennen, daß ein Chat-Werkzeug stets synchroner Natur ist. Leider ist meistens nur die Übermittlung von Text möglich, weshalb die Unterstützung nonverbaler Kommunikation sehr eingeschränkt ist. Wenn z.B. dem Gesprächspartner die eigene Stimmung vermittelt werden soll, so können auf Konventionen beruhende Zeichensequenzen eingegeben verwendet werden, die von dem Gesprächspartner als Stimmungsindikator gedeutet werden. Beispielhaft bedeutet die Zeichenfolge ":-)", daß man selbst schmunzeln muß bzw. sich freut, während ":-(" das Gegenteil ausdrückt. Auch bei einem Chat-Werkzeug ist die direkte Adressierung typisch. Eine Nachricht kann sich sowohl an einen konkreten oder aber an alle Gesprächspartner richten, jedoch kann sie nie von unbekanntem „Mithörern“ vernommen werden.

## 2.2.2 Koordinationsunterstützung

Die Koordination ist eine bei der kooperativen Aufgabenerledigung notwendige Tätigkeit, wie bereits in Abschnitt 2.1 gezeigt wurde. Malone und Crowstone (vgl. [MaCr90]) meinen, daß Koordination als eine eigenständige Aktivität betrachtet werden kann, die ein notwendiger Mehraufwand ist, wenn mehrere Parteien eine Aufgabe bearbeiten.

Hierbei dient die Koordination vordergründig dem gemeinsamen Ziel des Teams, indem gegenseitige Behinderungen der Kooperationspartner, Mehrfacherledigungen der Aufgaben oder gar Vernichtung von Ergebnissen vermieden werden. Burger bringt das wie folgt zum Ausdruck:

„Die Koordination eines Teams ist auf das Kooperationsziel hin ausgerichtet und sorgt dafür, daß dieses Ziel auf möglichst effizientem Weg erreicht wird und die benötigten kooperativen Tätigkeiten entsprechend optimal organisiert werden.“ ([Burg97], S. 43)

Charakteristisch für Kooperation ist, daß es zwischen den zu erledigenden kooperativen Tätigkeiten bestimmte Abhängigkeiten gibt. Somit besteht der Bedarf, daß diese Tätigkeiten zeitlich so angeordnet werden, daß sie ohne "Reibungsverluste" bearbeitet werden können. Malone und Crowstone sehen darin sogar das Wesen der Koordination.

„Coordination is managing dependencies between activities.“ ([MC94], s.90)

Neben den Abhängigkeiten zwischen den einzelnen Tätigkeiten kann es aber auch noch zu Situationen kommen, in denen konkurrierend auf gemeinsame Ressourcen zugegriffen wird. Hierin besteht eine weitere Aufgabe der Koordination, nämlich dem Auflösen von Konkurrenzsituationen.

Um diese "Reibungsverluste" aufgrund von Abhängigkeiten zwischen den Tätigkeiten zu minimieren sowie Konkurrenzsituationen zu vermeiden, sind bei der Koordination folgende Entscheidungen zu treffen (vgl. [Bur97], S. 43f.):

- Bestimmen und Aufteilen des gesamten Lösungsweges in Einzelschritte
- Zuordnung der einzelnen Tätigkeiten zu Teamteilnehmern (oder Rollen)
- zeitliche Ordnung von Tätigkeiten
- Zusammenführen von Ergebnissen

**Koordination:**

"Koordination ist der Prozeß oder der Mechanismus zur Abstimmung von Arbeitsteilung bei kooperativer Arbeit. Koordination kann auf wechselseitigen Konventionen oder ausdrücklichen Regeln beruhen." ([Zül98], S. 428)

Bei der Unterstützung der Koordination steht der prozeßorientierte Nutzen im Vordergrund, d.h. das Bestimmen der optimalen Reihenfolge von Einzeltätigkeiten gemäß ihrer Abhängigkeiten untereinander sowie das Auflösen und Vermeiden von Konkurrenzsituationen. Koordinationsunterstützende Software kann sich der Kommunikationsunterstützung bedienen bzw. mit integrieren, um u.a. eine Verständigung über die zu koordinierenden Tätigkeiten und Ressourcen zu ermöglichen. Beispiele für eine Koordinationsunterstützung sind der elektronische Kalender, Ressourcenplanung und Workflow-Management-Systeme. Während die koordinierende Funktion beim elektronischen Kalender sowie der Ressourcenplanung offensichtlich erscheint, möchte ich auf das Beispiel des Workflow-Management-Systems im folgenden detaillierter eingehen.

**Workflow-Management-System:**

Ein Workflow-Management-System ist ein Softwaresystem, das der Steuerung und Überwachung eines Workflows anhand eines Workflow-Schemas dient.

Hierbei werden unter einem Workflow (Arbeitsfluß) automatisiert ablaufende Teile eines Geschäftsprozesses verstanden. Wie ein Workflow zu steuern ist, wird in einem Workflow-Schema (Ablaufspezifikation) modelliert. Dieses Modell beschreibt die Reihenfolge, die Voraussetzungen sowie den Inhalt der von verschiedenen Bearbeitern auszuführenden Einzeltätigkeiten (vgl. [JBS97]). Ein besonderer Schwerpunkt bei Workflow-Management-Systemen ist die optimale Reihenfolge der Einzeltätigkeiten bei Geschäftsprozessen. Um eine optimale Reihenfolge zu erreichen, modelliert ein Experte, wie ein Geschäftsprozeß bearbeitet werden soll. Bei der Modellierung ist der Experte insbesondere in der Lage, Konkurrenzsituationen beim Zugriff auf gemeinsame Ressourcen zu erkennen und zu vermeiden.



Die Workflow-Management-Systeme haben den Charakter einer Koordinationsunterstützung, da sowohl die optimale Reihenfolge der Einzeltätigkeiten als auch das Auflösen von Konkurrenzsituationen anhand des Workflow-Schemas gesteuert werden.

### 2.2.3 Kooperationsunterstützung

Die beiden bisher beschriebenen Unterstützungsformen helfen dem Bearbeiter nicht immer optimal, wenn er kooperative Arbeit (siehe Abschnitt 2.1) zu erledigen hat. Das wird insbesondere dann offensichtlich, wenn zur Aufgabenerledigung situatives und kooperatives Arbeitshandeln charakteristisch sind. Der Bearbeiter kann zwar eine Kommunikationsunterstützung verwenden, doch ist diese nicht anwendungsfachlich motiviert, da sie lediglich dem Austausch beliebiger Informationen dient. Auch hilft ihm eine Koordinationsunterstützung nur bedingt weiter, da er sich zwar mit anderen Kooperationspartnern koordinieren muß, doch stellt Koordination nur einen Teil kooperativer Arbeit dar.

Zu diesem Zweck bietet sich eine spezielle Unterstützung an, die sich der Kommunikations- sowie Koordinationsunterstützung bedient (vgl. Abbildung 3). Diese stellt dem Bearbeiter Kooperationsmittel und -medien zur Verfügung, die eine arbeitsteilige Aufgabenerledigung erlauben. Gemäß dem WAM-Ansatz (siehe Kapitel 3) orientieren sich die Kooperationsmittel und -medien an Entwurfsmetaphern, die der zu unterstützenden Anwendungswelt entlehnt sind.

#### **Kooperationsmittel:**

"Ein Kooperationsmittel ist ein fachlich motivierter Gegenstand, der die Kooperation unterstützt. Er vergegenständlicht die Kooperation oder die dabei notwendige Koordination.

Beispiele für Kooperationsmittel sind Vorgangsmappen und Laufzettel. Eine Vorgangsmappe ermöglicht, Unterlagen in einem arbeitsteiligen Prozeß weiterzugeben. Laufzettel vergegenständlichen die Reihenfolge von kooperativen Arbeitsschritten und die jeweils Verantwortlichen." ([Zül98], S. 429)

#### **Kooperationsmedium:**

"Ein Kooperationsmedium ist ein fachlich motivierter Gegenstand, der zur Realisierung von Kooperation in Anwendungssystemen dient.

Gemeinsam ist allen Kooperationsmedien, daß mit ihrer Hilfe Materialien oder Information ausgetauscht werden können und daß sie selbst vergegenständlicht sind. Beispiele für Kooperationsmedien sind ein elektronisches Postversandsystem, Gruppenpostfächer oder ein elektronisches Notizbrett." ([Zül98], S. 429)

Eine Kooperationsunterstützung sollte auch die Art der Kooperation mit berücksichtigen. So wird zwischen der impliziten und der expliziten Kooperation unterschieden. Diese Unterscheidung wird getroffen, weil im Benutzungsmodell (siehe Abschnitt 3.2) der jeweiligen Groupware explizit die Kooperation und Koordination vorgesehen sein kann oder sich aufgrund der Nutzung gemeinsamer Ressourcen implizit ergibt. Sørgaard bringt das wie folgt zum Ausdruck:

"Two ways of coordinating cooperative work can be identified. One is by explicit communication about how the work is to be performed, another is less explicit, mediated by the *shared material* used in the work process." ([Sør88], S. 321)

Sørgaard führt in diesem Zitat den Begriff "Shared Material" ein, das von Gryczan (vgl. [Gry96]) als "gemeinsames Material" bezeichnet wird. Ich möchte an dieser Stelle diesen Begriff nicht weiter vertiefen, sondern nur festhalten, daß gemeinsame Materialien als Teilmenge von gemeinsamen Ressourcen anzusehen sind.

Wird aufgrund der gleichzeitigen Nutzung gemeinsamer Ressourcen kooperiert, so soll darunter die implizite Kooperation verstanden werden. Anwendungsfachlich bietet sich zur Bereitstellung von Kooperationsmittel und -medien das Konzept der Registratur (vgl. [Hav99], [GHR+00]) an.

#### **Implizite Kooperation:**

"Bei der impliziten Kooperation wird der konkurrierende Zugriff mehrerer Benutzer auf gemeinsame Ressourcen im Benutzungsmodell ermöglicht und verdeutlicht. Kooperation oder Koordination selbst sind aber nicht vergegenständlicht." ([Zül98], S. 430)

Folgendes Beispiel möchte ich hier skizzieren, um diesen Fall zu verdeutlichen.

Als Sachbearbeiter ist es u.a. meine Aufgabe, ein Dokument zu modifizieren, welches in einem Dokumentenverwaltungssystem zentral zugreifbar ist. Bei meinem Versuch, auf dieses Dokument zuzugreifen, stelle ich fest, daß es bereits durch einen anderen Bearbeiter für jegliche weitere fremde Zugriffe vorübergehend gesperrt wurde. Durch diesen Umstand bin ich zwar verhindert, meiner Aufgabe nachzugehen, doch schützt es mich davor, daß meine an diesem Dokument vorgenommenen Modifikationen unbeabsichtigt durch andere vernichtet werden könnten. Vielleicht erhalte ich auch noch durch das System die Information, wer der gegenwärtige Bearbeiter des von mir gewünschten Dokumentes ist. In diesem Fall kann ich mich direkt über entsprechende Kooperationsmittel und -medien an diesen Bearbeiter wenden, um mit ihm diese Konfliktsituation aufzulösen. Ich kann aber auch die Erledigung meiner Aufgabe solange aufschieben, bis das Dokument für mich wieder zugreifbar ist.

An diesem Beispiel wird deutlich, daß im Benutzungsmodell des Dokumentenverwaltungssystems nicht explizit vorgesehen ist, mit anderen Sachbearbeitern zu kooperieren. Vielmehr geschieht die Kooperation implizit aufgrund der Information, daß das gewünschte Dokument gerade von einem anderen Bearbeiter gesperrt ist.

Ist jedoch im Benutzungsmodell explizit vorgesehen, daß Benutzer eines Softwaresystems kooperieren müssen, um ihre Aufgabe erledigen zu können, so spricht man von expliziter Kooperation. Eine einfache Form der expliziten Kooperation ist die Kooperation mittels Materialaustausch.

#### **Explizite Kooperation:**

"Bei der expliziten Kooperation wird im Benutzungsmodell deutlich, daß mehrere Benutzer kooperativ in einer gemeinsamen Arbeitsumgebung arbeiten. Geeignete Kooperationsmittel und -medien stehen für die Weitergabe von Materialien und für die Koordination bereit." ([Zül98], S. 437)

Mit dem folgenden Beispiel möchte ich verdeutlichen, wie explizite Kooperation mittels Materialaustausch in Erscheinung treten kann.

Als Sachbearbeiter ist es meine Aufgabe, ein Dokument zu bearbeiten. Allerdings ist diesmal das Dokument nicht in einem Dokumentenverwaltungssystem zentral zugreifbar. Vielmehr muß ich das zu bearbeitende Dokument erst von einem anderen Sachbearbeiter, z.B. aus einer anderen Fachabteilung, anfordern. Zu diesem Zweck verfasse ich eine elektronische Notiz oder Anforderung und sende sie direkt zu meinem Kooperationspartner. Dieser wiederum sendet mir mit dem elektronischen Postversandssystem das gewünschte Dokument zu. Nachdem ich das Dokument in meinem Postkorb erhalten habe, befindet es sich in meinem privaten Arbeitsbereich und kann von mir gemäß meiner Aufgabe bearbeitet werden.

Wie in diesem Beispiel deutlich wird, ist es im Benutzungsmodell des vorhandenen Softwaresystems explizit vorgesehen, mit anderen Sachbearbeitern zu kooperieren, um eine Aufgabe erledigen zu können. Diese explizite Kooperation geschieht durch die Anforderung sowie den Austausch eines Dokumentes und wird durch ein elektronisches Postversandssystem unterstützt.

Eine Erweiterung der einfachen expliziten Kooperation mittels Materialaustausch ist die Vergegenständlichung routinierter Zusammenarbeit durch Prozeßmuster (vgl. [Wulf95], [Gry96], [Zül98], [Bre00]). Gryczan definiert das Prozeßmuster als:

"[...] ein gemeinsames Material zur Vergegenständlichung eines kooperativen Arbeitsprozesses. Durch das Prozeßmuster werden Verantwortlichkeiten von Personen oder Rollenträgern und Tätigkeiten in einem kooperativen Arbeitsprozeß festgelegt. Ein Prozeßmuster besteht aus der Angabe der Abhängigkeiten von und zwischen Tätigkeiten, die bei der kooperativen Arbeit zu erledigen sind und dazu notwendigen Dokumenten." ([Gry96], S. 178).

Das Konzept des Prozeßmusters kann durch die Kooperationsmittel *Vorgangsmappe und Laufzettel* realisiert werden. Hierbei stellt der Laufzettel die Vergegenständlichung eines kooperativen Arbeitsprozesses dar. Er schlüsselt auf, in welcher Abhängigkeit Einzeltätigkeiten mit den notwendigen Dokumenten auszuführen sind, um eine Aufgabe zu erledigen. Die Vorgangsmappe verkörpert den der Aufgabe zugehörigen Vorgang. Sie enthält alle Dokumente, die für die jeweiligen Tätigkeiten notwendig sind. Außerdem ist die Vorgangsmappe der Ort, an dem der Laufzettel angebracht ist.

Um einen Vorgang arbeitsteilig bearbeiten zu können, wird noch ein Kooperationsmedium benötigt, daß die Vorgangsmappe von einem Arbeitsplatz zu einem anderen befördert. Das in dieser Arbeit thematisierte Postversandssystem stellt ein solches Kooperationsmedium dar.

Im Rahmen dieser Arbeit werde ich das Konzept des Prozeßmusters nicht weiter vertiefen, da es in [Bre00] vor dem Hintergrund des gemeinsamen COJAC-Projektes (siehe Abschnitt 1.1) ausführlich erörtert wird.

Ein weiteres Konzept zur Unterstützung von Kooperation ist die erweiterte Raummetapher (vgl. [RoWo98], [Zül98] und [Koc00]). Gegenstand des Konzeptes ist, *Orte als Räume*<sup>2</sup> zu repräsentieren. In den Räumen können sich die Benutzer *gegenseitig wahrnehmen*<sup>3</sup> und miteinander kommunizieren. Obwohl vordergründig dieses Konzept der Kooperations- und Koordinationsunterstützung dient, hat es als weiteres wesentliches Merkmal, daß es auch zur *Orientierung* des Benutzers im System beiträgt. Insbesondere wird der Benutzer bei der

---

<sup>2</sup> Bei der Raummetapher wird Raum im Sinne von Zimmer verstanden.

<sup>3</sup> Das Konzept der gegenseitigen Wahrnehmung wird speziell in der englischsprachigen Literatur als "Awareness" bezeichnet.

Orientierung durch einen Raumplan unterstützt, der ihm eine Übersicht über alle vorhandenen Räume bietet.

Neben diesen primären Eigenschaften gibt es noch eine Reihe weiterer ([Zül98], S. 458ff). So wird bei der Raummetapher das Konzept der *Einheit von Ort und Zeit* realisiert, d.h. ein Material oder Werkzeug kann sich zu einem Zeitpunkt nur an einem Ort (Raum) befinden. Weiterhin hat die Raummetapher die Eigenschaft der *Abgeschlossenheit*. Das bedeutet, es gibt keine Orte außerhalb der Räume. Somit braucht der Benutzer die Dinge, die er nicht in den Räumen auffinden kann, auch nicht außerhalb zu vermuten. Damit ist schon eine weitere Fähigkeit erwähnt. In der Raummetapher ist die *Lokalisierung* von Dingen konzipiert, d.h. der Benutzer hat die Möglichkeit, nach den in den Räumen befindlichen Dingen zu suchen. Damit nicht für jeden Benutzer im Raumsystem alle Dinge zugreifbar sind, gibt es Konzepte für den Zugriffsschutz. Ein Konzept ist das der *Privatheit und Öffentlichkeit*, d.h. Räume können nur von speziellen oder von allen Benutzern "betreten" werden. Wie es bei einem konkreten Raum der Fall ist, hängt davon ab, wie die Rechte für den "Zutritt" unter den Benutzern verteilt sind. Ein weiteres Konzept für den Zugriffsschutz ist das *Handgepäck*. Jeder Benutzer hat sein privates Handgepäck, das sich aus verschiedenen Dingen, wie z.B. Materialien, zusammensetzt. Dieses Handgepäck ist für den Benutzer immer zugreifbar, egal in welchem Raum er sich befindet. Jedoch bleibt es allen anderen Benutzern verborgen.

Da Räume Dinge enthalten können, sind sie auch in der Lage, mit geeigneten Kooperationsmedien ausgestaltet zu werden. Da z.B. das Postversandsystem zur Vergegenständlichung als Kooperationsmedium Postkörbe anbietet (siehe Kapitel 5), läßt es sich zur Ausstattung von Räumen verwenden. Dadurch wird speziell die Kooperation zwischen Privaträumen der Benutzer unterstützt, was sonst aufgrund des Zugriffsschutzes unmöglich wäre.

Im Rahmen dieser Arbeit werde ich das Konzept der Raummetapher nicht weiter vertiefen, da es in [Koc00] vor dem gemeinsamen Hintergrund des COJAC-Projektes (siehe Kapitel 1) umfassend diskutiert wird.

Einige Beispiele zur Unterstützung kooperativer Arbeit wurden bereits genannt. So ist das Postversandsystem mit Postkörben eine Möglichkeit explizite Kooperation zu realisieren. Weiterhin ist die gebotene Funktionalität des Postversandsystems so elementar, daß es sich auch als Grundlage für weitere Softwaresysteme zur Kooperationsunterstützung verwenden läßt.

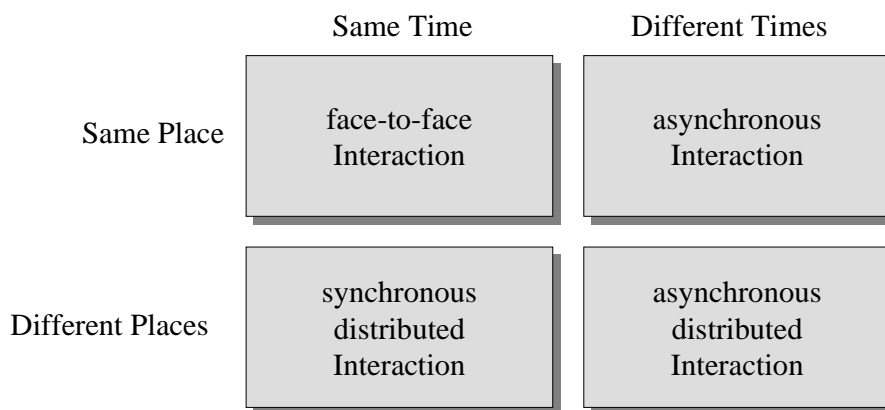
Implizite Kooperation wird z.B. durch eine Registratur unterstützt (vgl. [Hav99], [GHR+00]). In einer Registratur können Materialien abgelegt und wieder entnommen werden. Sie stellt sicher, daß ein Material nicht mehrfach in einer Systemumgebung vorhanden ist. Vielmehr ist es dem Benutzer nur möglich, wenn bereits das Original entnommen wurde, sich Kopien geben zu lassen. Mit einer Kopie kann aber niemals ein abgelegtes Original ersetzt werden. Wichtig für die Unterstützung der impliziten Kooperation ist, daß dem Benutzer mitgeteilt wird, wenn das angeforderte Original nicht zugreifbar ist, von wem es entnommen wurde. Ob sich nun der Benutzer mit dem Entnehmer des Originals verständigt, ist nicht im Benutzungsmodell der Registratur explizit vorgesehen.

### 2.3 Time-Space-Matrix

Zur Unterstützung der vielen Ausprägungen kooperativer Arbeit gibt es eine breite Palette an Groupware, die auf den jeweiligen Anwendungskontext zugeschnitten ist. Abbildung 4 zeigt eine mögliche Klassifizierung.

Wie in dieser Abbildung zu erkennen ist, wird nicht zwischen Kommunikations-, Koordinations- und Kooperationsunterstützung unterschieden, sondern es wird danach differenziert, ob sich die kooperierenden Personen zur gleichen Zeit am gleichen Ort befinden oder nicht. Ellis et al. betonen, daß neben der Orts- und Zeitgleichheit sicherlich noch weitere Merkmale zur stärkeren Differenzierung zwischen den einzelnen Sparten von Groupware denkbar sind (vgl. [EGR91]). Im folgenden werde ich aber von einer weiteren Differenzierung absehen.

Bei der Betrachtung der Orts- und Zeitgleichheit der Aufgabenerledigung der Kooperationspartner ergeben sich vier mögliche Konstellationen (siehe Abbildung 4), die ich im folgenden kurz erläutern werde.



**Abbildung 4: Time-Space-Matrix ([EGR91], S. 41)**

#### **Face-to-face Interaction (synchrone ortsgleiche Interaktion)**

Als Grundlage für die synchrone ortsgleiche Interaktion<sup>4</sup> verstehe ich, daß sich die Kooperationspartner in unmittelbarer Nähe aufhalten und Blickkontakt haben. Da sie sich an dem gleichen Ort befinden, können sie den Kooperationspartner sowie seine Interaktionen zeitgleich wahrnehmen. Das versetzt die Kooperierenden in die Lage, sofort auf Aktionen des Partners zu reagieren sowie über das Handeln implizit zu kooperieren. Ein Beispiel für besonders stark ausgeprägte implizite ortsgleiche Kooperation ist ein Fußballspiel. Ein Spieler einer Mannschaft kann anhand der Positionen seiner Mitspieler sowie dem aktuellen Ballbesitz erkennen, wohin er als nächstes zu laufen und was er dort zu tun hat. Ein weiteres Beispiel ist ein Gespräch von Angesicht zu Angesicht, bei dem u.a. auch gestikuliert wird.

Es ist offensichtlich, daß sich nicht viele Situationen, in denen synchron interagiert wird, sinnvoll mit dem Computer unterstützen lassen. Durch die gegebenen Rahmenbedingungen,

<sup>4</sup> Interaktion wird im Rahmen dieser Arbeit als interpersonale Kommunikation verstanden ([Bro90], S. 211).

d.h. gleicher Ort sowie gleiche Zeit, kommt es kaum zu Einschränkungen in der Kommunikationsvielfalt. Deshalb kann hier eine Computerunterstützung vielmehr vorhandene Möglichkeiten ausbauen, vereinfachen sowie effektiver gestalten, nicht aber Einschränkungen beheben.

Als Computerunterstützung für diese Art der Interaktion sind Konferenzsysteme zu nennen, die ein gemeinsames Material allen Konferenzmitgliedern, die sich gemeinsam in einem Raum befinden, zur Verfügung stellen. Dieses Material, z.B. eine gemeinsame Zeichenfläche, fungiert als Gegenstand, der Diskussionspunkte zu verdeutlichen hilft.

### **Synchronous distributed Interaction (synchrone verteilte Interaktion)**

Bei der synchronen verteilten Interaktion kommunizieren zwar die Kooperationspartner zeitgleich, doch befinden sie sich an unterschiedlichen Orten. Dadurch wird es notwendig, Informationen zum Kooperationspartner zu transportieren. Weiterhin hat das gegenüber der synchronen ortsgleichen Interaktion einige Einschränkungen der Kommunikationsmöglichkeiten zur Folge, da die visuelle Wahrnehmung entfällt. Dadurch ist es auch nicht mehr möglich, eine Vielzahl an Ereignissen zu vernehmen, die dem Kooperationsumfeld und nicht direkt dem Kooperationspartner entstammen. Weiterhin kann auch nicht mehr uneingeschränkt in die Kooperationssituation eingegriffen werden.

Ein klassisches Beispiel für eine synchrone verteilte Interaktion ist das Telefonieren. Die Gesprächspartner können zeitgleich miteinander reden, ohne sich dabei in unmittelbarer Nähe zu befinden. Allerdings kann jegliche Gestik sowie Mimik des anderen nicht mehr vernommen werden, so daß eine Vielzahl an zusätzlichen Informationen verloren gehen, die neben der expliziten auch eine implizite Kommunikation gestatten würden. Das soll aber nicht heißen, daß alle impliziten Informationen abhanden kommen. So kann anhand der Betonung der Aussprache durchaus noch die Gefühlslage des Gesprächspartners am Telefon vernommen und entsprechend darauf reagiert werden. Es können allerdings keine direkten Eingriffe in die gegenwärtige Kooperationssituation vorgenommen werden. Teilt z.B. der Gesprächspartner mit, daß er mit dem Auto auf der Autobahn eine Panne hat, so kann über das Telefon unmöglich direkt an dieser Situation etwas geändert werden.

Ein Beispiel für die Computerunterstützung synchroner verteilter Interaktion sind sogenannte Konferenzsysteme, die Konferenzen zwischen mehreren Teilnehmern an verschiedenen Orten gestatten. Diese Systeme können unterschiedlich viel Komfort den Teilnehmern bieten, wie z.B. die Übertragung von Text- oder von Videoinformationen.

### **Asynchronous Interaction (asynchrone ortsgleiche Interaktion)**

Zwar befinden sich bei der asynchronen ortsgleichen Interaktion die Kooperationspartner in unmittelbarer Nähe, doch erfolgt der Informationsaustausch zeitlich versetzt. Hierfür wird ein Informationsträger verwendet. Auf diesem können Informationen hinterlassen werden, die später von dem Kooperationspartner wieder eingeholt werden. Weiterhin stellt der Informationsträger auch den Ort der Interaktion dar. Aufgrund des asynchronen Charakters der Interaktion ist eine unmittelbare Reaktion auf Ereignisse nicht mehr möglich. Es ist nicht mehr vorhersehbar, wann der Kooperationspartner die Information erhält bzw. einholt. Das hat nicht nur Einfluß auf die mögliche Kooperationsvielfalt, sondern auch auf den Inhalt der dabei ausgetauschten Informationen.

Eine Wandtafel, an der Notizzettel angeheftet werden können, ist ein Beispiel für die asynchrone ortsgleiche Interaktion. Der Ort, an dem die Interaktionen von statten gehen, ist für alle Kooperationspartner immer der gleiche und wird durch die Wandtafel selbst repräsentiert. Wann allerdings eine angebrachte Notiz gelesen wird, bleibt unklar. Deshalb macht es keinen Sinn, über eine Wandtafel Informationen auszutauschen, auf die der Kooperationspartner sofort reagieren müßte.

Beispiele für eine Computerunterstützung sind Newsgroups im Internet sowie sogenannte Whiteboards oder aber auch im System öffentliche Ordner, in denen Nachrichten abgelegt werden können.

### **Asynchronous distributed Interaction (asynchrone verteilte Interaktion)**

Bei der asynchronen verteilten Interaktion tauschen die Kooperationspartner von verschiedenen Orten zu beliebigen Zeitpunkten Informationen miteinander aus. Diese Art der Interaktion bietet die größte Flexibilität, da es keine Anforderungen an die Rahmenbedingungen Ort und Zeit gibt. Allerdings bedeutet das auch die größten Einschränkungen in den Einsatzmöglichkeiten, da sich bei dieser Art die Nachteile aufgrund der Zeit- sowie der Ortsverschiedenheit (siehe in den entsprechenden vorherigen Abschnitten) kumulieren.

Beispielhaft für asynchrone verteilte Interaktion ist das Verfassen und Senden von Briefen mit der Post. Der Vorteil, in dieser Art und Weise zu interagieren, besteht darin, daß sich die Kooperationspartner an unterschiedlichen Orten zu verschiedenen Zeiten aufhalten können. In den Vorteilen sind aber auch wiederum die Nachteile begründet. Da der Kooperationspartner sich nicht am gleichen Ort befindet, kann er keine Kontextinformationen vernehmen, die nicht Inhalt des Informationsaustausches sind. Weiterhin ist auch unklar, wann er eine gesendete Nachricht erhält und liest.

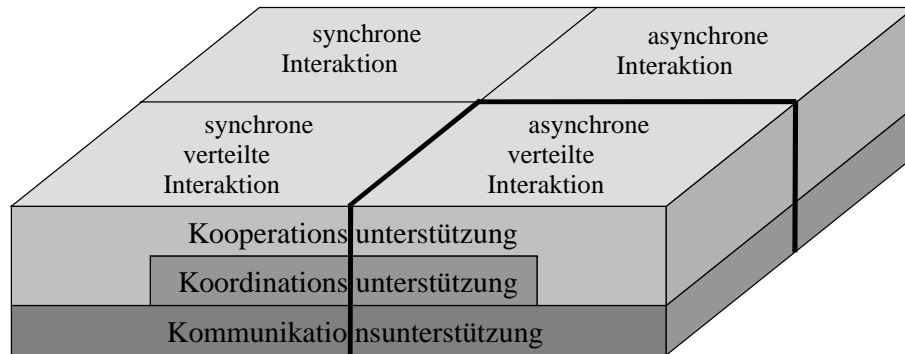
Ein Beispiel für eine mögliche Computerunterstützung ist die elektronische Post. Dieses Medium gestattet den Transport von beliebigen Daten, die in einem festgelegten Format adressiert sein müssen, binnen kürzester Zeit. Es können nicht nur Informationen in Form von Text versendet werden, sondern auch Programme oder sonstige Daten wie z.B. Grafiken und Tabellen.

In den vier erläuterten Konstellationen bezüglich der Orts- und Zeitgleichheit wird deutlich, daß eine sinnvolle Computerunterstützung von der konkreten zu unterstützenden Kooperationssituation abhängig ist. Es ist also nicht nur von Bedeutung, ob eine Computerunterstützung die Kooperation, die Koordination oder die Kommunikation fokussiert, sondern auch, welche Rahmenbedingungen in der jeweiligen Situation gegeben sind.

## **2.4 Zusammenfassung und Ziel**

In den vorangegangenen Abschnitten 2.2 und 2.3 sind verschiedenen Aspekte aufgezeigt, die bei einer Klassifizierung der Kooperationsunterstützung relevant sind. So ist zum einen die Art der kooperativen Tätigkeit (Kooperation, Koordination bzw. Kommunikation) als auch

deren Rahmenbedingungen (Orts- und Zeitgleichheit) von Bedeutung. Es fällt auf, daß die einzelnen Aspekte sich orthogonal zueinander verhalten und somit einen Raum aufspannen, in dem eine ganz konkrete Unterstützung plaziert werden kann. Die Abbildung 5 veranschaulicht diesen Sachverhalt.



**Abbildung 5: Würfel der Unterstützung**

Da es den Rahmen dieser Arbeit sprengen würde, für jeden in Abbildung 5 erkennbaren Bereich eine Unterstützung anzubieten, habe ich mich auf den Bereich, für den asynchrone verteilte Interaktionen charakteristisch sind, beschränkt (in Abbildung 5 durch einen schwarzen Rand hervorgehoben). Die Kooperationsform, die hier angesiedelt ist, ist die explizite Kooperation mittels Materialaustausch (siehe Abschnitt 2.2.3). Meine Fokussierung darauf ist so zu begründen, daß sowohl die technische Unterstützung dieser Kooperationsform den größten Nutzen verspricht, als auch die meisten Kooperations-situationen diese Eigenschaften aufweisen. Rodden und Blair beschreiben das wie folgt:

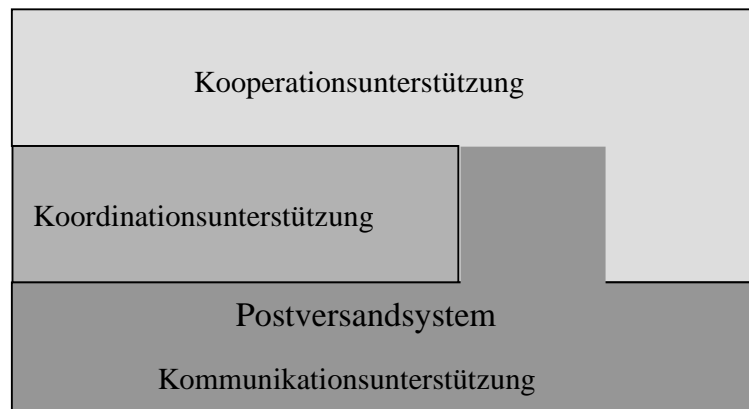
„[...] mailing capability [...] has become accepted as a standard means of cooperation in a distributed system and provides adequate support for the development of asynchronous cooperative systems.“ ([RoBI91], S. 54)

Mit dieser Ausrichtung möchte ich ein Postversandsystem schaffen, das sowohl die Kommunikation als auch die Kooperation unterstützt (in Abbildung 6 dunkel hervorgehoben) und der Koordination als Grundlage dient. Hinsichtlich der Kommunikationsunterstützung wird es die Funktionalität anbieten, Materialien asynchron zwischen verschiedenen Arbeitsplätzen auszutauschen. Diese Funktionalität wird mit einer kooperationsunterstützenden Komponente ergänzt, die eine Differenzierung der Materialien sowie deren anwendungsfachlich motivierten Umgang erlaubt und die Funktionalität des Materialaustauschs über ein Kooperationsmedium vergegenständlicht. Dieses Kooperationsmedium kann flexibel in verschiedenen Anwendungskontexten eingesetzt werden und auch als Basis für weitere unterstützende Softwaresysteme, z.B. zur Realisierung von Prozeßmustern, dienen. Weiterhin wird das Postversandsystem Kooperationsmittel anbieten, die situatives arbeitsteiliges Arbeitshandeln begünstigen.

Ziel ist es nicht, ein Postversandsystem zu entwickeln, das nur für ausgewählte Kooperations-situationen gedacht ist, wie z.B. eine herkömmliche elektronische Post. Vielmehr sollen solche Programme das von mir entwickelte Postversandsystem als Grundlage nutzen können.



Das Postversandsystem wird zweierlei Charaktere haben, um den genannten Anforderungen gerecht zu werden. Zum einen wird es direkt einsetzbar sein, da alle Komponenten implementiert sind, die für die Unterstützung der verschiedenen Benutzungsmodelle aus Abschnitt 4.3 notwendig sind. Da es als Mehrbenutzer-Software, die zur Unterstützung kooperativer Arbeit dient, genutzt werden kann, erfüllt es alle Eigenschaften von Groupware (siehe Abschnitt 2.1). Zum anderen bietet es Schnittstellen an, welche die Anpassung sowie die Benutzung einzelner Komponenten gestatten.



**Abbildung 6: Einordnung des Postversandsystems**

Mit diesen Vorgaben für das Postversandsystem möchte ich die Zielsetzung aus Abschnitt 1.2 erneut aufgreifen und wie folgt konkretisieren.

Das Ziel dieser Arbeit ist die Konzeption und die Realisierung eines Postversandsystems im Kontext des WAM-Ansatzes als flexibel kombinier- und anpaßbare **Basiskomponenten zur Unterstützung expliziter asynchroner verteilter Kooperation mittels Materialaustausch** und dessen **Integration ins JWAM-Rahmenwerk**.



### 3 WAM-Ansatz

Das Postversandsystem ist in dem Kontext des WAM-Ansatzes eingebettet. Er ist die Grundlage der Ideen und Konzepte dieser Arbeit. Aus diesem Grund widmet sich dieses Kapitel dem WAM-Ansatz und vermittelt einen Überblick.

Der WAM-Ansatz wird am Arbeitsbereich Softwaretechnik der Universität Hamburg in Forschung und Lehre vertreten. Seit der Einrichtung dieses Arbeitsbereiches im Jahre 1991 wurde dieser Ansatz sukzessive weiterentwickelt. So sind in diesem Rahmen eine Reihe von Publikationen (vgl. [GKZ94], [Gry96], [Zül98]) sowie Studien- und Diplomarbeiten entstanden.

Dieses Kapitel gliedert sich in folgende Abschnitte:

Abschnitt 3.1 beschreibt das Wesen des WAM-Ansatzes. Hier zeige ich kurz auf, wie der WAM-Ansatz den Softwareentwicklungsprozeß unterstützt. Insbesondere gehe ich darauf ein, wie auf der Grundlage von objektorientierten Entwurfs- und Konstruktionstechniken mit Hilfe eines Leitbilds mit Entwurfsmetaphern eine enge Korrespondenz zur Softwarearchitektur geschaffen wird.

Abschnitt 3.2 beschäftigt sich dann eingehender mit den Leitbildern. Besonderen Wert lege ich in diesem Abschnitt auf das Leitbild vom *Arbeitsplatz für eigenverantwortliche Expertentätigkeit*, da es für die Konzeption des Postversandsystems den Rahmen bildet.

Abschnitt 3.3 behandelt die Entwurfsmetaphern. Dabei erläutere ich, wie diese ein konkretes Leitbild ausgestalten und helfen, es in ein konkretes Softwaresystem zu überführen. Im weiteren Verlauf stelle ich die verschiedenen Entwurfsmetaphern aus dem WAM-Repertoire vor.

#### 3.1 WAM-Ansatz als Softwareentwicklungsmethode

Der WAM-Ansatz verkörpert eine anwendungsorientierte und menschenzentrierte Herangehensweise bei der Softwareentwicklung mit dem Ziel, dem Benutzer eine Unterstützung zur Aufgabenerledigung zu schaffen. Mit dem Ziel verbindet sich aber auch, daß die Software vom Benutzer akzeptiert wird. Diese Akzeptanz kann nur durch einen hohen Gebrauchswert für den Benutzer erreicht werden. Der Gebrauchswert wird u.a. durch folgende Eigenschaften bestimmt:

- " - Die Funktionalität orientiert sich an den Aufgaben aus dem Anwendungsbereich.
- Die Handhabung des Systems ist benutzergerecht.
- Die im System festgelegten Abläufe und Schritte lassen sich je nach Anwendungssituation problemlos an die tatsächlichen Anforderungen anpassen." ([Zül98], S. 4)

Zu diesem Zweck vereinigt der WAM-Ansatz ein Leitbild mit Entwurfsmetaphern, anwendungsorientierte Dokumente sowie eine objektorientierte und evolutionäre partizipative Vorgehensweise mit Prototyping. Softwareentwicklung wird als ein wechselseitiger Kommunikations- und Lernprozeß verstanden.

Um einen hohen Grad an Anwendungsorientierung sowie Akzeptanz zu erreichen, bilden die Gegenstände, Begriffe sowie Konzepte des zu unterstützenden Anwendungsbereiches die Grundlage beim Softwareentwurf. Die daraus resultierende *Strukturähnlichkeit* schafft eine Grundlage für die gemeinsame Kommunikation zwischen Entwickler und Benutzer. Außerdem ermöglicht sie es dem Benutzer sowohl das Softwaremodell besser zu verstehen, als auch die entwickelte Software intuitiver zu bedienen. Durch das bessere Verständnis des Softwaremodells kann der Benutzer die Entwicklung der Anwendungssoftware stärker beeinflussen. Dadurch wird erreicht, daß die Software den Anforderungen entspricht.

Die Verwendung der Gegenstände, Begriffe sowie Konzepte der Anwendungswelt beim Entwurf sichert allein noch keine Anwendungsorientierung zu, wenn nicht die Verwendung aufgrund einer expliziten Sichtweise erfolgt. Sie hilft sowohl den Entwicklern als auch den Benutzern beim Entwurf und Verständnis der Software. Um eine explizite Sichtweise zu schaffen, werden im WAM-Ansatz *Leitbilder* verwendet (vgl. Abschnitt 3.2). Das gebräuchlichste Leitbild ist *der Arbeitsplatz für eigenverantwortliche Expertentätigkeit*.

Ein Leitbild kann durch passende *Entwurfsmetaphern* ausgestaltet werden (vgl. Abschnitt 3.3). Sie verbildlichen Komponenten und Konzepte des Anwendungssystems mit den Gegenständen des Anwendungsbereiches und bauen somit die Grundlage für eine gemeinsame Kommunikation zwischen Benutzer und Entwickler aus.

Darüber hinaus haben die Entwurfsmetaphern neben ihrer Bildhaftigkeit auch einen technischen Hintergrund in Form von Konstruktionsanleitungen und Entwurfsmustern<sup>5</sup>. In Verbindung mit dem oben genannten Leitbild des Arbeitsplatzes für eigenverantwortliche Expertentätigkeit werden beim WAM-Ansatz die Entwurfsmetaphern Werkzeug, Automat, Material und Umgebung verwendet.

Weiterhin unterstützen *anwendungsorientierte Dokumente* den Entwicklungsprozeß. Diese lassen sich in folgende Dokumenttypen einteilen:

- *Szenarios* beschreiben die aktuellen Situationen des Anwendungsbereiches, die für eine Unterstützung relevant sind.
- Ein *Glossar* definiert die Fachbegriffe des Anwendungsbereiches.
- *Systemvisionen* beschreiben, wie ein Anwendungssystem die Aufgaben der Benutzer unterstützen soll.
- *Entwurfsdokumente* werden von Entwicklern erstellt, um die Softwarearchitektur in den verschiedenen Phasen zu dokumentieren. Hierbei kommen u.a. Klassen-, Objekt- und Interaktionsdiagramme zum Einsatz.

Die Verwendung dieser Dokumenttypen ist prinzipiell nicht neu. Das Wesentliche im Kontext des WAM-Ansatzes ist, daß die Dokumente mit einem anwendungsfachlich motivierten Sprachgebrauch erstellt werden.

Ein weiterer Schwerpunkt, um einen hohen Gebrauchswert durch die Anwendungsorientierung zu erreichen, ist die *Gestaltung des Entwicklungsprozesses*. Im WAM-Ansatz wird deshalb eine evolutionäre Vorgehensweise mit Prototyping favorisiert (vgl. [Flo84]). Grundlegend für diese Vorgehensweise ist, daß der Entwicklungsprozeß als Kommunikations- und Lernprozeß verstanden wird, bei dem die Beteiligten während der

---

<sup>5</sup> Entwurfsmuster werden in der englischsprachigen Literatur häufig "Design Pattern" genannt.

gesamten Entwicklungsdauer die Konzepte und Entwürfe diskutieren, rückkoppeln und weiterentwickeln. Von großer Bedeutung bei dieser Herangehensweise ist der Einsatz von Prototypen (vgl. [Flo84], [PoBl96]). Sie verkörpern und veranschaulichen den Entwurf des zukünftigen Anwendungssystems.

### 3.2 Leitbilder

Ein Problem bei der Entwicklung von Anwendungssoftware ist, daß die Entwickler eine Vision von deren Gestaltung und Handhabung haben müssen. Zu diesem Zweck erstellen sie Benutzungsmodelle.

#### **Benutzungsmodell:**

"Ein Benutzungsmodell ist ein fachlich orientiertes Modell darüber, wie Anwendungssoftware bei der Erledigung der anstehenden Aufgaben im jeweiligen Einsatzkontext benutzt werden kann.

Das Benutzungsmodell umfaßt eine Vorstellung von der Handhabung und Präsentation der Software aber auch von den fachlichen Gegenständen, Konzepten und Abläufen, die von der Software unterstützt werden.

Es ist sinnvoll, ein Benutzungsmodell auf der Grundlage eines Leitbilds mit Entwurfsmetaphern zu realisieren." ([Zül98], S. 71)

Das Benutzungsmodell sollte in sich stimmig sein, so daß man von einer Einheit von Inhalt und Form der Anwendungssoftware reden kann. Um diese Stimmigkeit herzustellen, wird ein Orientierungsrahmen gebraucht, der den Entwicklern eine gemeinsame gestalterische Sicht und Sprache gibt, um über den Anwendungsbereich und dessen Unterstützung diskutieren zu können. Dieser Orientierungsrahmen soll durch Leitbilder geschaffen werden.

Allgemein ist ein Leitbild eine grundsätzliche Sichtweise, anhand derer ein Ausschnitt der Realität wahrgenommen, verstanden und gestaltet werden kann (vgl. [Zül98], S. 73). In der Softwareentwicklung hat ein Leitbild sowohl konstruktive als auch analytische Funktion. Die konstruktive Funktion läßt sich daran erkennen, daß ein Leitbild den Entwicklern helfen soll, das fachliche Modell des Anwendungsbereiches in ein Anwendungssystem zu überführen. Analytisch kann ein Leitbild verwendet werden, um bereits bestehende Anwendungssoftware vergleichen und bewerten zu können. Außerdem hilft es dem Benutzer, die entwickelte Software bei der Benutzung besser zu verstehen.

#### **Leitbild in der Softwareentwicklung:**

„Ein Leitbild in der Softwareentwicklung gibt im Entwicklungsprozeß und für den Einsatz einen gemeinsamen Orientierungsrahmen für die beteiligten Gruppen.

Es unterstützt den Entwurf, die Verwendung und die Bewertung von Software und basiert auf Wertvorstellungen und Zielsetzungen.

Ein Leitbild kann konstruktiv oder analytisch verwendet werden.“ ([Zül98], S. 73)

So verschieden, wie die zu unterstützenden Anwendungsbereiche vom Charakter her sein können, so verschieden können auch die Leitbilder sein, die den Entwicklern und Benutzern als Orientierung dienen. Es liegt nahe, daß entsprechend der Merkmale des

Anwendungsbereiches ein Leitbild gewählt werden sollte. In Abbildung 7 sind exemplarisch vier Leitbilder aufgeführt.

Leitbild	Gestaltungsziel	Rolle der Anwender	Rolle der Entwickler
Objektwelten	Objektorientierung unmittelbar auf den Entwurf übertragen	Impulsgeber, der die Objekte aktiviert	Schöpfer von Miniwelten
Direkte Manipulation von Arbeitsgegenständen	bekannte Arbeitsgegenstände selbstverständlich manipulieren	Akteur, Bearbeiter	Konstrukteur von Artefakten
Fabrik	menschliche Arbeit automatisieren und kontrollieren	Maschinenbediener, Störfaktor	Maschinenbauer, Maschineneinrichter
Arbeitsplatz für eigenverantwortliche Expertentätigkeit	qualifizierte Arbeit durch geeigneten Arbeitsplatz unterstützen	eigenverantwortlicher Experte, der Fachsprache spricht	Werkzeugbauer, Arbeitsplatzgestalter

**Abbildung 7: Leitbilder in der Softwareentwicklung ([Zül98], S. 74)**

Im WAM-Ansatz wird als Leitbild der *Arbeitsplatz für eigenverantwortliche Expertentätigkeit* vertreten. Bei diesem Leitbild treten die Benutzer als Experten ihres Anwendungsbereiches auf, die genug Wissen und Erfahrung haben, selbst zu entscheiden, wie sie ihre Aufgaben erledigen. Dazu bedienen sie sich verschiedener Hilfsmittel, die sie der gegenwärtigen Situation entsprechend auswählen. Vor dem Hintergrund dieses Leitbildes kann eine Unterstützung in der Art erfolgen, daß dem Benutzer den Aufgaben angemessene Software-Artefakte zur Verfügung gestellt werden. Diese geben ihm keine Handlungsabläufe vor, sondern erlauben ihm ein situatives menschliches Arbeitshandeln.

Diese Betrachtung von Anwendungssystemen wird als *unterstützende Sichtweise* bezeichnet. Bei dieser Sichtweise steht die qualifizierte menschliche Arbeit im Vordergrund und gilt als unersetzlich. Im Gegensatz dazu steht die *ablaufsteuernde Sichtweise*, in der die Qualifikation der Benutzer von untergeordneter Bedeutung ist und die Abläufe bei der Aufgabenerledigung fest vorgegeben werden.

### 3.3 Entwurfsmetaphern

Entwurfsmetaphern sind bildhafte Vorstellungen von Gegenständen und Konzepten des Anwendungsbereiches und gestalten damit ein Leitbild aus. Sie schaffen ein gemeinsames Verständnis sowie eine Kommunikationsgrundlage zwischen den Entwicklern und Benutzern. Aufgrund ihrer Eigenschaft, ein Leitbild auszugestalten, leiten sie die Vorstellung, wie ein Leitbild in ein Anwendungssystem überführt werden kann.

Die Wahrnehmung sowie die Sicht auf den Anwendungsbereich wird bei der Verwendung von Entwurfsmetaphern beeinflusst, d.h. es wird das Wesentliche für das Anwendungssystem

hervorgehoben und in einen Zusammenhang gebracht. Maß und Oberquelle beschreiben dies wie folgt:

„Looking at some things from a certain perspective means highlighting certain aspects while hiding others, means seeing certain relations and not others. [...] To someone who has a hammer everything looks like a nail. So the application of some particular perspective helps one to see more clearly in one respect while blinding to other respects.“ ([MO92], S. 234)

Durch diesen Einfluß auf die Wahrnehmung des Anwendungsbereiches fördern Entwurfsmetaphern das Verständnis von Funktionalität und Handhabung des Anwendungssystems und tragen zu dessen Gestaltung bei.

Neben den bisher beschriebenen Eigenschaften haben Entwurfsmetaphern immer auch einen technischen Hintergrund in Form von Konstruktionsanleitungen und Entwurfsmustern. Für Gamma et al. sind Entwurfsmuster:

"[...] descriptions of communicating objects and classes that are customized to solve a general design problem in a particular context." ([GHJ+95], S. 3)

Aufgrund der bisher beschriebenen Eigenschaften von Entwurfsmetaphern definiere ich sie wie folgt (vgl. [Zül98], S. 79):

#### **Entwurfsmetapher**

Eine Entwurfsmetapher verbildlicht Gegenstände und Konzepte des Anwendungsbereiches und gestaltet damit ein Leitbild aus. Sie prägt die analytische Sicht auf den Anwendungsbereich und hat immer eine technisch konstruktive Interpretation.

Da für die weitere Argumentationsfolge die bereits erwähnten Entwurfsmuster von Bedeutung sind, komme ich zu folgender Definition:

#### **Entwurfsmuster:**

Ein Entwurfsmuster beschreibt eine Entwurfslösung für ein allgemeines Problem in einem konkreten Kontext. Es besteht aus den Elementen:

- Name
- Problembeschreibung
- Lösung in Form von einer Architektur aus Objekten und Klassen
- Konsequenzen

Das im WAM-Ansatz vordergründig vertretenen Leitbild vom *Arbeitsplatz für eigenverantwortliche Expertentätigkeit* wird durch die Entwurfsmetaphern Werkzeug, Material, Automat und Umgebung konkretisiert. Diese sind im folgenden im Kontext einer Computerunterstützung thematisiert.

#### **Werkzeug**

Ein Werkzeug vergegenständlicht ein Hilfsmittel, mit dem Materialien bearbeitet, sondiert und dargestellt werden können. Es unterstützt den Benutzer bei der Erledigung immer wiederkehrender Aufgaben. Dabei gibt es dem Benutzer nicht die einzelnen Arbeitsschritte vor, sondern bietet ihm Funktionalität zur freien Auswahl an. Der Benutzer kann eigenverantwortlich und situativ entscheiden, in welcher Reihenfolge er die einzelnen Arbeitsschritte ausführt, um zu seinem angestrebten Ziel zu gelangen.

Neben der Funktionalität muß auch die Handhabung des Werkzeuges der Aufgabenerledigung angemessen sein. Ein Werkzeug arbeitet stets reaktiv, d.h. die Aktionen zur Bearbeitung des Materials werden vom Benutzer interaktiv ausgelöst.

### **Material**

Ein Material vergegenständlicht ein fachlich motiviertes Arbeitsmittel, das Teil des Arbeitsergebnisses ist. Es ist nicht in der Lage, sich selbst dem Benutzer zu repräsentieren, sondern es wird immer von einem Werkzeug dargestellt.

Änderungen an einem Material sind ausschließlich nur mittels Werkzeug oder Automat möglich. Daher muß das Material für die Bearbeitung geeignet sein.

Ein Material sollte keine generischen sondern nur fachlich motivierte Umgangsformen an seiner Schnittstelle anbieten. Damit meine ich, daß der spezifische Umgang an der Schnittstelle des Materials erkennbar sein sollte. Die Umgangsformen sollten nicht von solch allgemeiner Natur sein, daß sie zu vielen verschiedenen Materialien passen würden.

Es ist möglich, daß ein Gegenstand des Anwendungsbereiches sowohl als Material als auch als Werkzeug verwendet werden kann. Seine Art der Verwendung ist aber in einer konkreten Situation immer eindeutig aus dem Verwendungszusammenhang erkennbar. So kann z.B. ein Hammer als Werkzeug dienen, um einen Nagel in die Wand zu schlagen. Andererseits stellt er ein Material dar, wenn dessen Griff repariert wird.

### **Automat**

Mitunter ist der Benutzer bei der Aufgabenerledigung mit als lästig empfundenen Routinetätigkeiten konfrontiert. Diese kann er zwar mit Werkzeugen ausführen, jedoch wäre eine automatisierte Unterstützung hilfreicher. Genau für diesen Zweck gibt es Automaten.

Ein Automat realisiert einen fachlichen Algorithmus, der mit einer vorbestimmten Folge von Arbeitsschritten ein fest definiertes Ergebnis liefert. Er kann einzelne aber auch eine Vielzahl verschiedenartiger Materialien bearbeiten.

Einmal konfiguriert und gestartet, kann der Automat über einen längeren Zeitraum hinweg ohne äußere Einwirkungen arbeiten. Die Konfiguration eines Automaten geschieht mit Hilfe von Einstellwerkzeugen.

Nachdem ein Automat gestartet wurde, arbeitet er unauffällig im Hintergrund. Es ist jederzeit möglich, seinen Zustand zu überprüfen, um u.a. festzustellen, ob er seine Aufgabe korrekt erledigt oder ein unerwünschtes Fehlverhalten aufweist.

### **Umgebung**

Mit Werkzeug, Material und Automat wurden bisher die Arbeitsmittel definiert, die von einem Anwendungssystem zur Verfügung gestellt werden. Nur selten kommt eine umfassende Unterstützung mit nur einem Werkzeug und einem Material aus. Hieraus ergibt sich die Notwendigkeit, den Ort für alle zugreifbaren Arbeitsmittel zu vergegenständlichen. Aus diesem Grund wurde die Entwurfsmetapher Umgebung von Gryczan eingeführt:

„Eine Arbeitsumgebung zur Unterstützung qualifizierter menschlicher Tätigkeiten ist der Ort für eine anwendungsfachlich motivierte Zusammenstellung von Werkzeugen, Automaten und Materialien. Durch die Arbeitsumgebung werden keine Reihenfolgebedingungen für die Verwendung von Werkzeugen und Arbeitsmitteln festgelegt.“ ([Gry96], S. 138)



Allerdings muß der Begriff Umgebung differenzierter betrachtet werden. So gibt es den *Arbeitsplatz*, der den Ort der unmittelbaren Aufgabenerledigung darstellt, und die *Arbeitsumgebung*, in der sich alle weiteren Arbeitsmittel zugreifbar befinden.

Neben der Eigenschaft, daß der Arbeitsplatz der Ort der Aufgabenerledigung ist, gewährt er dem Benutzer eine Privatsphäre. Dadurch ist sichergestellt, daß in einer kooperativen Arbeitsumgebung niemand Einblick und Zugriff auf die eigenen Arbeitsmittel hat. Für Burger (vgl. [Bur97]) ist diese Privatsphäre Voraussetzung dafür, daß die Benutzer das Anwendungssystem akzeptieren und einsetzen. Weiterhin erlaubt der Arbeitsplatz eine den Bedürfnissen des Benutzers entsprechende Anordnung der Arbeitsmittel.

Geht es bei einem Anwendungssystem nur um die Unterstützung eines Einzelarbeitsplatzes, so ist eine Differenzierung zwischen Arbeitsplatz und Arbeitsumgebung nicht weiter notwendig und die Begriffe können synonym verwendet werden.



## 4 Anforderungen an das Postversandsystem

Für das Postversandsystem gibt es zwei verschiedene Klassen von Anforderungen, die unterschieden werden müssen. Die erste Klasse stellt die fachlichen Anforderungen dar. Sie beschreiben, wie der Anwender bei seiner Aufgabenerledigung unterstützt werden soll. Die zweite Klasse sind die technischen Anforderungen. Sie definieren, wie das Postversandsystem für verschiedene Anwendungskontexte eingesetzt werden kann und wie es sich in andere Softwaresysteme integrieren läßt. Sie sind vorrangig für die Softwareentwickler von Interesse, die das Postversandsystem verwenden wollen.

Um sowohl die fachlichen als auch die technischen Anforderungen zu erarbeiten, gliedert sich dieses Kapitel in die folgenden Abschnitte:

Abschnitt 4.1 befaßt sich mit allgemeinen fachlichen Anforderungen und kontrastiert sie gegenüber den Eigenschaften einer herkömmlichen elektronischen Post.

Abschnitt 4.2 geht auf die technischen Anforderungen ein, die einen Rahmen für die Realisierung bilden.

In Abschnitt 4.3 werden als Teil der fachlichen Anforderungen drei Benutzungsmodelle skizziert, die das Postversandsystem unmittelbar und ohne Anpassungen unterstützen soll. Diese orientieren sich am Leitbild vom Arbeitsplatz für eigenverantwortliche Expertentätigkeit.

Abschnitt 4.4 widmet sich als spezielle technische Anforderung den Eigenschaften, welche die Softwarearchitektur des Postversandsystems aufweisen muß, damit es als eine Basisunterstützung für kooperative Arbeit in verschiedenen Anwendungskontexten wiederverwendet werden kann.

### 4.1 Fachliche Anforderungen

Charakteristisch für die Unterstützung der expliziten Kooperation durch das Postversandsystem ist der *Austausch von Materialien* zwischen verschiedenen Arbeitsplätzen. Dabei verstehe ich die Materialien im Sinne des WAM-Ansatzes (siehe Abschnitt 3.3).

Für das Postversandsystem ist es unerheblich, um was für konkrete Materialien es sich handelt. So können es z.B. Texte, Bilder und Personendaten sein.

Darüber hinaus ist von großer Bedeutung, daß sich die Materialien direkt versenden lassen, ohne erst in ein anderes allgemeines Datenformat konvertiert werden zu müssen. Dadurch wird dem Empfänger die Rückkonvertierung erspart, die zu Fehlern führen kann. So könnte es z.B. dem Empfänger passieren, daß er Bilddaten erhält, diese aber in einen Text konvertiert.

Ein weiterer Vorteil ist, wenn sich Materialien direkt versenden lassen, daß die verschiedenen Materialien differenziert behandelt werden können und somit einen anwendungsfachlich motivierten Umgang erlauben, insbesondere beim Versenden und Empfangen. So können z.B. beim Versenden von Vorgangsmappen Informationen gesammelt werden, die eine nachträgliche Vorgangsverfolgung erlauben (vgl. [Bre00]). Ein weiteres Beispiel ist, daß empfangene Materialien anhand eines anwendungsfachlich motivierten Kriteriums, wie deren Materialtyp oder Absender, automatisch in verschiedene Behälter abgelegt werden.

Im Gegensatz zum Postversandsystem gestattet eine herkömmliche elektronische Post lediglich den Austausch von Textnachrichten. An diese Textnachrichten können Dateien angehängt werden, um auch Daten anderen Formates mitzusenden. Jedoch bedeutet das, daß die Daten immer erst in einer Datei gespeichert werden müssen, damit sie versendet werden können.

Mit den beschriebenen Möglichkeiten einer herkömmlichen elektronischen Post verbinden sich auch eine Reihe von Unzulänglichkeiten. So liegt es z.B. in der Verantwortung des Senders, den Dateinamen so zu wählen, daß der Empfänger mit der Datei etwas anzufangen weiß. Weiterhin ist es nicht möglich, Nachrichten anhand ihres Typs zu differenzieren, da es nur Textnachrichten gibt. Somit können auch keine Automatismen geschaffen werden, die anwendungsfachlich motiviert mit den Nachrichten umgehen, wie z.B. das Ablegen der Nachrichten in verschiedene Behälter.

Es stellt sich nun die Frage, was es bedeutet, wenn ein Material versendet bzw. empfangen wird. Wirft man einen Blick in die Anwendungswelt, so scheint die Antwort recht trivial. Wenn ein Material versendet wurde, so ist es vom eigenen Arbeitsplatz verschwunden und befindet sich irgendwo anders, möglicherweise schon beim Empfänger. Wenn ein Material empfangen wurde, dann befindet es sich auf dem eigenen Arbeitsplatz und nirgendwo sonst. Etwas abstrakter ausgedrückt, kann sich ein Material zu einem Zeitpunkt immer nur an einem Ort befinden. In diesem Zusammenhang spricht man auch von der *Einheit von Ort und Zeit*. Daraus ergibt sich eine weitere wesentliche Anforderung an das Postversandsystem, nämlich daß die Einheit von Ort und Zeit gewahrt wird.

Eine Konsequenz aus dieser Anforderung ist, daß ein Material nicht an mehrere Empfänger zeitgleich gesendet werden kann. Um dennoch einen ähnlichen Umgang dem Benutzer zu ermöglichen, wäre ein Konzept von Original und Kopie notwendig. Dieses könnte so gestaltet sein, daß nur ein explizit bestimmter Empfänger das Original erhält, während den anderen Kopien zugesandt werden. Eine solche Kopie stellt ein eigenständiges Material dar und unterscheidet sich zumindest in der Identifikation vom Original.

Neben dem besseren Verständnis des Benutzers für das, was es bedeutet, ein Material zu versenden bzw. zu empfangen, vereinfacht sich auch die Verwaltung der Materialien. Denn ein Material kann nicht zeitgleich mehrfach an verschiedenen Orten, wie z.B. Postkörben, abgelegt werden, so daß zu einem späteren Zeitpunkt evtl. unklar ist, welches den aktuellen Bearbeitungsstand repräsentiert.

Die Forderung nach der Einheit von Ort und Zeit vereinfacht nicht nur den Umgang mit den Materialien, sondern ermöglicht auch deren systemweite Lokalisierung. Zu diesem Zweck wird eine Vorgangsverfolgung (vgl. [Bre00]) verwendet. Diese „beobachtet“, wie Materialien zwischen den Arbeitsplätzen mit dem Postversandsystem ausgetauscht werden. Somit kann die Vorgangsverfolgung u.a. befragt werden, wo sich ein bestimmtes Material befindet.

Bei den herkömmlichen elektronischen Postsystemen gibt es dieses Konzept von der Einheit von Ort und Zeit nicht. Folglich kann sich eine Textnachricht samt Anhang beliebig oft im Gesamtsystem sowie auf dem Arbeitsplatz befinden. Es wird dadurch nicht nur der Verwaltungsaufwand höher, um eine unkontrollierte Vervielfältigung zu vermeiden und somit Speicherkapazitäten zu verschwenden, sondern es ist auch schwer nachzuvollziehen, welches Exemplar den aktuellen Bearbeitungsstand verkörpert.

Weiterhin bietet eine herkömmliche elektronische Post die auf den ersten Blick sehr nützliche Möglichkeit, eine Textnachricht samt Dateianhang an beliebig viele Empfänger gleichzeitig zu senden. Da es aber kein Konzept von Original und Kopie gibt, kann sich diese Fähigkeit als sehr tückisch erweisen. Wenn z.B. ein Vorgang repräsentiert durch eine Vorgangsmappe als Dateianhang über die herkömmliche elektronische Post an mehrere Empfänger gesendet wird, ist unklar, welches Exemplar der weiteren Bearbeitung dienen soll. Konfliktsituationen sind somit vorprogrammiert. Weiterhin läßt sich auch nachträglich nur sehr schwer eindeutig dieser Vorgang verfolgen, um den aktuellen Bearbeitungsstatus zu erfahren.

Ein weiteres Problem ist, daß das Postversandsystem in verschiedenen Anwendungskontexten einsetzbar sein soll. Daraus ergibt sich die Notwendigkeit, daß es *verschiedene Benutzungsmodelle* unterstützen muß. Welches Benutzungsmodell das Postversandsystem aktuell hat, soll konfigurierbar sein. Ein Beispiel hierfür ist, daß einem Benutzer ein einzelner Posteingangskorb ausreichen würde, wenn er nur wenige Materialien empfängt. Dagegen kann es für einen Benutzer sehr wichtig sein, wenn er Empfänger einer Vielzahl verschiedener Materialien ist, diese nach bestimmten Kriterien sortiert auf mehrere Posteingangskörbe aufzuteilen. Auf die verschiedenen Benutzungsmodelle, die das Postversandsystem unterstützt, gehe ich in Abschnitt 4.3 detailliert ein.

Bei einer herkömmlichen elektronischen Post ist das Benutzungsmodell fest vorgegeben. Es lassen sich zwar oftmals noch einige Einstellungen vornehmen, doch kann durch die Konfiguration die Handhabung nur bedingt an den Anwendungskontext angepaßt werden. Es ist meistens nicht möglich, die Software mit eigenen an den Anwendungskontext spezialisierten Komponenten zu erweitern, um dem gewünschten Benutzungsmodell zu entsprechen.

Wie bereits in Abschnitt 2.4 erwähnt, soll das Postversandsystem die *asynchrone verteilte Kooperation* unterstützen. Um das zu erreichen, stellt es dem Benutzer Kooperationsmittel und -medien (siehe Abschnitt 2.2.3) zur Verfügung, derer er sich frei bedienen kann. Um das Kooperationsmedium zu vergegenständlichen, das dem Austausch von Materialien zwischen verschiedenen Arbeitsplätzen dient, werden Postkörbe verwendet. Als Kooperationsmittel werden Transportbehälter angeboten, in die der Benutzer die zu sendenden Materialien ablegen kann. Ein weiteres Kooperationsmittel stellen die Vorgangsmappen dar, die ein Teil des Ergebnisses der Arbeit von Breitling (vgl. [Bre00]) sind.

Bei der Verwendung der Kooperationsmittel sowie -medien ist von Bedeutung, daß dem Benutzer keine Handlungsabfolgen vorgegeben werden. Dadurch bleibt ihm die Flexibilität erhalten, eigenverantwortlich zu entscheiden, wie er das Postversandsystem, insbesondere die Kooperationsmittel sowie -medien, in den verschiedenen Anwendungssituationen nutzt.

Eine herkömmliche elektronische Post unterstützt dagegen lediglich die asynchrone verteilte Kommunikation. Das bedeutet, daß ihre Fähigkeiten, die über den Austausch von Textnachrichten hinausgehen, beschränkt sind. Dieses Defizit ging z.T. aus den bisher beschriebenen Anforderungen hervor.

Dem Benutzer steht in der Regel bei der herkömmlichen elektronischen Post ein Anwendungsprogramm zur Verfügung, welches das Verfassen, Absenden und Anzeigen von Textnachrichten unterstützt. Leider ist dessen Handhabung fest vorgegeben, die sich nur teilweise dem Anwendungskontext anpassen läßt. So können z.B. empfangene Nachrichten sortiert nach Absender in verschiedene Ordner abgelegt werden, jedoch kann das Kriterium sich nicht anwendungsfachlich motiviert auf den Inhalt beziehen.

Weiterhin kann der Benutzer bei der herkömmlichen elektronischen Post nur Textnachrichten mit Dateianhang versenden. Ihm stehen jedoch keine weiteren alternativen Kooperationsmittel zur Verfügung, die dem Anwendungskontext angepaßt sein können. Somit ist eine optimale anwendungsbezogene Kooperationsunterstützung nicht möglich.

Abbildung 8 faßt noch einmal die genannten Anforderungen an das Postversandsystem zusammen und stellt diese einer herkömmlichen elektronischen Post gegenüber.

Postversandsystem	elektronische Post
asynchrone verteilte Kooperation mit Hilfe von Kooperationsmitteln und -medien	asynchrone verteilte Kommunikation mit Hilfe eines Anwendungsprogrammes
verschiedenste Materialtypen können zwischen den Arbeitsplätzen ausgetauscht werden	nur Textnachrichten mit Dateianhang können versendet werden
Einheit von Ort und Zeit der Materialien	keine Einheit von Ort und Zeit der Materialien
Unterstützung verschiedener Benutzungsmodelle	Benutzungsmodell ist fest vorgegeben

**Abbildung 8: Anforderungen an das Postversandsystem im Vergleich zu einer herkömmlichen elektronischer Post**

Ich möchte an dieser Stelle noch einmal darauf hinweisen, daß wegen der aufgezeigten fachlichen Anforderung an das Postversandsystem eine herkömmliche elektronische Post nicht als verwerflich zu betrachten ist. Sie weist lediglich für die Kooperationsunterstützung bei arbeitsteiligen Arbeitshandlungen Unzulänglichkeiten auf. Das bedeutet aber nicht, daß sie in vielen Kooperationssituationen keine Existenzberechtigung hat. Sie kann durchaus neben dem Postversandsystem verwendet werden, jedoch muß dem Benutzer klar sein, wann er welches System benutzen sollte.

## 4.2 Technische Anforderungen

In Abschnitt 2.4 hatte ich u.a. als Ziel formuliert, daß das Postversandsystem als Grundlage für kooperationsunterstützende Software eingesetzt werden kann. Daraus erwächst die Anforderung, daß das Postversandsystem *Basisfunktionalität* anbietet. Unter Basisfunktionalität verstehe ich Funktionalität, die so elementar und abstrakt gestaltet ist, daß sie in vielerlei Softwaresystemen mit unterschiedlichen Anwendungskontexten verwendet werden kann. Da die Softwarearchitektur des Postversandsystems eine Reihe von Eigenschaften aufweisen muß, um Basisfunktionalität dem Entwickler anzubieten, gehe ich in Abschnitt 4.4 detailliert darauf ein.

Eine weitere technische Anforderung an das Postversandsystem ist, wie bereits in Abschnitt 2.4 gefordert, daß es als *Groupware* einsetzbar ist. Damit beziehe ich mich insbesondere auf die Eigenschaft, daß das Postversandsystem als Mehrbenutzer-Software zur Unterstützung kooperativer Arbeit nutzbar ist.

Wie bereits in Kapitel 1 erwähnt, hat das Postversandsystem als Kontext den WAM-Ansatz. Dieser Ansatz wird durch das an der Universität Hamburg im Fachbereich Softwaretechnik entwickelte JWAM-Rahmenwerk technisch unterstützt. Daraus ergibt sich die zusätzliche technische Anforderung, daß das Postversandsystem integraler *Bestandteil des JWAM-Rahmenwerkes* werden soll (siehe Kapitel 6).

### **4.3 Benutzungsmodelle**

Wie bereits im Abschnitt 4.1 als fachliche Anforderung erwähnt wurde, soll das Postversandsystem in der Lage sein, eine Vielzahl verschiedener Benutzungsmodelle zu unterstützen. Es soll möglich sein, daß das Benutzungsmodell dem jeweiligen Anwendungskontext angepaßt werden kann.

Anknüpfend an das Leitbild vom Arbeitsplatz für kooperative eigenverantwortliche Expertentätigkeit habe ich mich exemplarisch für die folgenden drei Benutzungsmodelle entschieden.

Das erste Benutzungsmodell (siehe Abschnitte 4.3.1) beschreibt die Mindestvoraussetzungen und den minimalen Umgang des Benutzers mit dem Postversandsystem beim Versenden und Empfangen von Materialien.

Im zweiten (siehe Abschnitt 4.3.2) wird dargestellt, wie die Benutzung des Postversandsystems beim Versenden und Empfangen entsprechend dem Anwendungskontext ausgestaltet bzw. erweitert werden kann.

Das dritte (siehe Abschnitt 4.3.3) hat einen ganz anderen Charakter. Es beschreibt die Voraussetzungen und den Umgang eines Benutzers in der Rolle eines Administrators, der das Postversandsystem konfiguriert. Die Notwendigkeit dieses Benutzungsmodells resultiert im wesentlichen aus der Anforderung, daß das Postversandsystem je nach Konfiguration verschiedene Benutzungsmodelle unterstützen soll.

#### **4.3.1 Einfacher Umgang**

Dieses Benutzungsmodell ist so rudimentär, daß es zwar die Kooperation unterstützt, aber in der Anwendungswelt sehr schnell an seine Grenzen stoßen würde. In Abbildung 9 ist es grafisch veranschaulicht.

*Voraussetzung:*

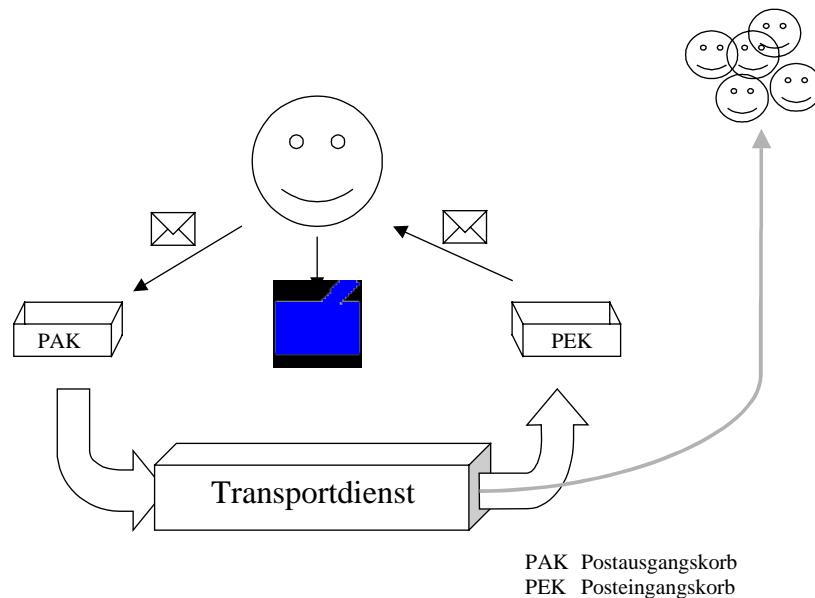
Der Benutzer hat zwei Behälter, die als Postein- und als Postausgangskorb fungieren, sowie ein Adressierungswerkzeug (siehe Abschnitt 5.4.1) auf seinem Arbeitsplatz<sup>6</sup>. Weiterhin hat er Zugriff auf ein Standardwerkzeug für Behälter<sup>7</sup>. Hiermit kann er sich den Inhalt eines Behälters, wie z.B. eines Postkorbes, anzeigen lassen und Materialien entnehmen. Dieses Standardwerkzeug ist nicht in der Abbildung 9 dargestellt, da es nicht Bestandteil des Postversandsystems ist und als gegebene Grundausstattung des Arbeitsplatzes vorausgesetzt wird.

---

<sup>6</sup> Der Arbeitsplatz wird im JWAM-Rahmenwerk durch den Desktop in Form einer grafischen Oberfläche visualisiert (vgl. [Lip99]).

<sup>7</sup> Das Standardwerkzeug für Behälter gehört zur Grundausstattung eines mit dem JWAM-Rahmenwerk entwickelten Arbeitsplatzes.

Dem Benutzer ist bewußt, daß es einen Transportdienst gibt, der die zu versendenden Materialien aus dem Postausgangskorb entnimmt bzw. die empfangenen in den Posteingangskorb ablegt. Weiterhin weiß er, daß nur versendbare Materialien, also die, die auch mit einer Empfängeradresse versehen werden können, vom Transportdienst übermittelt werden.



**Abbildung 9: Einfaches Benutzungsmodell**

*Versenden:*

Bevor der Benutzer ein versendbares Material versenden kann, muß er es für einen Empfänger adressieren. Dazu verwendet er das Adressierungswerkzeug. Anschließend legt der Benutzer das zu versendende Material mittels Drag-and-Drop<sup>8</sup> in den Postausgangskorb. Der Transportdienst entnimmt in periodischen Abständen den Inhalt des Postausgangskorbes und versucht die Materialien den Empfängern zuzustellen. Für den Benutzer ist der Transportdienst transparent. Er bemerkt ihn lediglich dadurch, daß der Postausgangskorb ohne eigenes Zutun entleert wird. Um festzustellen, ob die in den Postausgangskorb abgelegten Materialien versendet wurden, kann der Benutzer sich dessen Inhalt mit dem Standardwerkzeug für Behälter anzeigen lassen. Prinzipiell kann er sogar noch nicht versendete Materialien wieder aus dem Postausgangskorb entnehmen und verhindert somit deren Versendung.

*Empfangen:*

Das Empfangen von Materialien geschieht für den Benutzer völlig verborgen. Er kann den Empfang lediglich so feststellen, indem er mit dem Standardwerkzeug für Behälter sich den Inhalt des Posteingangskorbes anschaut. Hat er ein Material empfangen, so kann er es mit dem Standardwerkzeug aus dem Posteingangskorb entnehmen.

<sup>8</sup> Drag-and-Drop ist eine Umgangsart mit den Werkzeugen und Materialien auf dem Desktop ([Lip99], S. 51ff.).

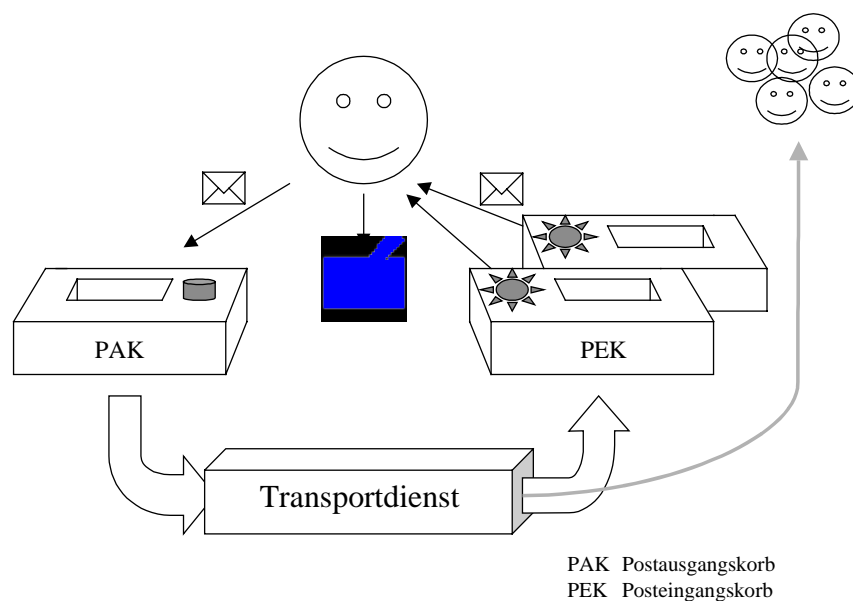


### *Fehlersituationen:*

Kann ein Material nicht dem gewünschten Empfänger zugestellt werden, so wird es vom Transportdienst in seinem Posteingangskorb abgelegt. Hat der Benutzer jedoch das zu versendende Material vergessen zu adressieren, so verbleibt es im Postausgangskorb.

### 4.3.2 Erweiterter Umgang

Dieses Benutzungsmodell setzt auf dem oben beschriebenen einfachen Umgang auf und erweitert es. Die Erweiterung besteht im wesentlichen darin, daß es dem Benutzer mehr Flexibilität bei der Verwendung der Postkörbe erlaubt.



**Abbildung 10: Erweitertes Benutzungsmodell**

### *Voraussetzung:*

Der Ausgangspunkt ist der gleiche wie beim einfachen Umgang (siehe Abschnitt 4.3.1) nur mit folgenden Erweiterungen. Der Benutzer hat nicht nur genau einen Postein- und Postausgangskorb, sondern es können beliebig viele sein. Die Postkörbe können verschiedene Eigenschaften aufweisen. So gibt es z.B. Posteingangskörbe, die nur für Materialien von speziellen Absendern eingerichtet sind. Die Art und Anzahl der verschiedenen Postkörbe sowie deren Einstellungen werden durch den Administrator konfiguriert (siehe Abschnitt 4.3.3).

### *Versenden:*

Das Versenden funktioniert wie beim einfachen Umgang (siehe Abschnitt 4.3.1) beschrieben, jedoch kommen folgende Erweiterungen hinzu. Der Postausgangskorb signalisiert visuell über sein Icon<sup>9</sup>, ob sich Materialien in ihm befinden. Dadurch kann der Benutzer schon anhand des Icons erkennen, ob der Inhalt des Postausgangskorbes durch den Transportdienst

<sup>9</sup> Alle Materialien und Werkzeuge werden auf dem Desktop durch Icons, welche kleine grafische Bilder sind, dargestellt.

versendet wurde. Er muß nun nicht mehr mit dem Standardwerkzeug für Behälter den Postausgangskorb einsehen, um festzustellen, ob sich noch etwas darin befindet.

Gegenüber dem einfachen Umgang kann der Benutzer über ein Postausgangskorbwerkzeug verfügen. Dieses Werkzeug bietet ihm die Funktionalität, daß er den Versandvorgang des Transportdienstes manuell starten kann. Dadurch kann er den Zeitpunkt bestimmen, wann ein Material versendet wird.

#### *Empfangen:*

Das Empfangen funktioniert wie in Abschnitt 4.3.1 beschrieben, jedoch kommen folgende Erweiterungen hinzu.

Die Posteingangskörbe haben die Fähigkeit, wie die Postausgangskörbe auch, anhand ihres Icons zu visualisieren, ob sie leer sind oder nicht. Dadurch muß sich der Benutzer nicht erst den Inhalt des Posteingangskorbes anzeigen lassen, um zu sehen, ob er etwas empfangen hat.

Zur Anzeige des Inhaltes eines Posteingangskorbes steht ihm ein spezielles Werkzeug zur Verfügung. Es zeigt ihm neben den empfangenen Materialien auch die jeweiligen Absender an.

Da der Benutzer in diesem Benutzungsmodell mehrere Posteingangskörbe mit verschiedenen Eigenschaften haben kann, werden nicht mehr alle empfangenen Materialien zwangsläufig in ein und denselben Posteingangskorb vom Transportdienst abgelegt. Hat er z.B. einen Posteingangskorb, der für einen speziellen Absender eingerichtet ist, so werden alle von diesem Absender empfangenen Materialien in genau diesem Posteingangskorb vom Transportdienst abgelegt. Dadurch kann der Benutzer schon anhand des entsprechenden Icons erkennen, ob er etwas von einem bestimmten Absender zugesendet bekommen hat.

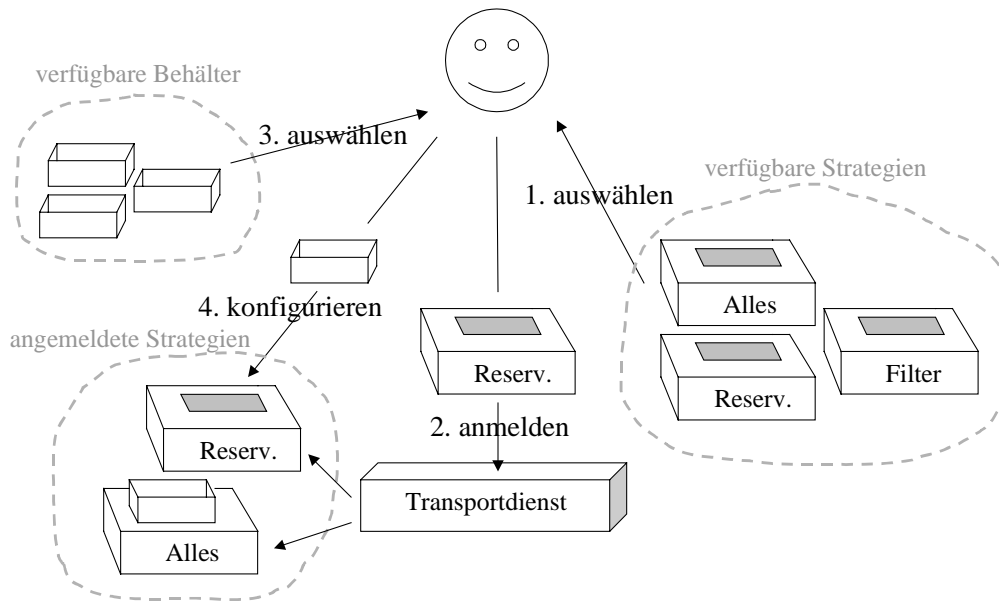
Aufgrund der Tatsache, daß es speziell eingerichtete Posteingangskörbe geben kann, muß sich auf dem Arbeitsplatz des Benutzers noch ein Standardposteingangskorb befinden. In diesem Posteingangskorb werden all die empfangenen Materialien abgelegt, die nicht einem anderen zugeordnet werden konnten.

#### *Fehlersituationen:*

Die Fehlersituationen, die auftreten können, sind die gleichen wie beim einfachen Umgang (siehe Abschnitt 4.3.1). Ich sehe dafür auch keine andere Handhabung vor. Jedoch sind hier Erweiterungen denkbar.

### 4.3.3 Konfiguration

Aus der Anforderung zur Unterstützung mehrerer Benutzungsmodelle resultiert, daß das Postversandssystem konfigurierbar sein muß. Die Konfiguration wird von einem Administrator vorgenommen. Ich möchte hier betonen, daß auch der Benutzer als Administrator fungieren kann. Abbildung 11 stellt das Benutzungsmodell zur Konfiguration des Posteingangs grafisch dar.



**Abbildung 11: Benutzungsmodell der Konfiguration des Posteingangs**

*Voraussetzung:*

Der Administrator weiß, daß es keine vorgefertigten Postkörbe gibt. Ihm ist bewußt, daß man einen Behälter als Postkorb nutzen kann, wenn er als solcher beim Transportdienst registriert ist.

Weiterhin bietet das Postversandsystem zwei spezielle Behälter an, die sich aufgrund ihrer Icons besonders gut als Postein- bzw. Postausgangskörbe eignen. Diese Behälter sind nämlich fähig, den Füllstand mit Hilfe ihrer Icons visuell darzustellen.

Um einen Behälter als Posteingangskorb zu konfigurieren, kann der Administrator auf verschiedene Posteingangsstrategien zugreifen, die ihm das Postversandsystem anbietet. Es stehen z. Zt. eine Reservierungs- und eine Standardstrategie<sup>10</sup> zur Verfügung. Weiterhin kann er ein spezielles Einstellwerkzeug (siehe Abschnitt 5.4.4) nutzen, daß ihm bei der Konfiguration unterstützt.

Um einen Behälter als Postausgangskorb nutzen zu können, stehen dem Administrator Postausgangsautomaten zur Verfügung. Z. Zt. gibt es lediglich einen einfachen Postausgangsautomaten, da dieser bereits den Anforderungen der zuvor beschriebenen Benutzungsmodelle gerecht wird.

Dem Administrator ist bewußt, daß er mit der Auswahl und der Zuordnung der Posteingangsstrategien sowie des Postausgangsautomaten zu den Behältern das Benutzungsmodell des Anwenders determiniert.

*Konfigurieren:*

Bevor der Administrator beginnen kann, Behälter als Postkörbe zu konfigurieren, muß er sicherstellen, daß sich die Behälter auf dem Arbeitsplatz des Benutzers befinden. Sollte dem nicht so sein, so muß er diese zuerst erzeugen.

<sup>10</sup> Es können beliebig viele weitere Strategien hinzugefügt werden, um den jeweiligen Anwendungskontext optimal zu unterstützen.

Um Behälter als Posteingangskörbe zu konfigurieren, kann der Administrator prinzipiell wie folgt vorgehen. Die genaue Handhabung bei der Verwendung des entsprechenden Einstellwerkzeuges ist in Abschnitt 5.4.4 beschrieben.

1. Er startet das Einstellwerkzeug, welches ihm alle verfügbaren Posteingangsstrategien anzeigt. Weiterhin kann der Administrator in dem Werkzeug sehen, welche Strategien bereits angemeldet wurden. Er wählt nun eine verfügbare Posteingangsstrategie aus.
2. Mit Hilfe des Einstellwerkzeuges meldet der Administrator die ausgewählte Posteingangsstrategie beim Transportdienst an. Eine erfolgreiche Anmeldung kann der Administrator daran erkennen, daß die Posteingangsstrategie in der Liste der angemeldeten Strategien aufgenommen wurde.
3. Falls es notwendig ist, daß die angemeldete Posteingangsstrategie konfiguriert werden muß, startet der Administrator ein entsprechendes Sub-Einstellwerkzeug. Dieses zeigt dem Administrator eine Liste aller verfügbaren Behälter an, mit denen er die Strategie einrichten kann. Aus dieser Liste wählt er einen Behälter aus.
4. Wenn der Administrator eine Auswahl getroffen hat, dann bestätigt er seine Eingabe. Daraufhin wird die Posteingangsstrategie mit dem ausgewählten Behälter vom Sub-Einstellwerkzeug eingerichtet.

Die Konfiguration der Postausgangskörbe kann analog zu der oben geschilderten Vorgehensweise geschehen. Jedoch werden anstatt der Posteingangsstrategien die Postausgangsautomaten verwendet.

#### **4.4 Architektureigenschaften**

Um den Softwareentwickler die Verwendung der von dem Postversandssystem angebotenen Basisfunktionalität zu erleichtern, ist es erforderlich, daß es eine Softwarearchitektur aufweist, die eine generische Lösung für explizite Kooperation mittels Materialaustausch repräsentiert und einen Rahmen für die Verwendung vorgibt. Eine solche Architektur wird als *Rahmenwerk* bezeichnet (siehe Abschnitt 4.4.1).

Syri fordert im folgenden Zitat von einem Rahmenwerk, daß es eine Konfiguration auf verschiedenen Ebenen erlauben sollte:

"A suitable framework has to allow configuration on different levels - for the system designer, the system administrator as well as for the end-user - to adapt the system to the working environment." ([Syri97], S. 157)

Die Eigenschaften, die in diesem Abschnitt erläutert werden, beziehen sich auf die Konfigurationsebene des Softwareentwicklers. Sie sollen sicherstellen, daß das Postversandssystem oder Teile dessen mit minimalem Aufwand in einer Vielzahl von Anwendungskontexten eingesetzt werden können. Die anderen beiden Konfigurationsebenen aus obigem Zitat wurden bereits im Abschnitt 4.3 berücksichtigt.

##### **4.4.1 Rahmenwerk**

Wie ich oben bereits bemerkt habe, sollte die Softwarearchitektur des Postversandsystems dem eines Rahmenwerkes entsprechen. Gamma et al. definieren den Begriff Rahmenwerk als:

"[...] a set of cooperating classes that make up a reusable design for a specific class of software." ([GHJ+95], S. 26)

Darin kommt zum Ausdruck, daß ein Rahmenwerk nicht nur mehrere Klassen enthält, die eine Funktionalität anbieten, sondern auch eine Softwarearchitektur.

Bei der Verwendung eines Rahmenwerkes zum Bau einer Anwendung werden nicht nur dessen Klassen sondern auch die gesamte Architektur wiederverwendet. Damit gibt ein Rahmenwerk die Konstruktion einer Anwendung vor. Der Bau einer Anwendung geschieht durch die Spezialisierung einzelner Klassen des Rahmenwerkes bzw. durch deren Benutzung.

**Rahmenwerk:**

Ein Rahmenwerk stellt eine allgemeine Entwurfslösung für einen Problembereich dar. Es ist eine Softwarearchitektur bestehend aus mehreren Klassen, die in einer vorgegebenen Beziehung zueinander stehen.

Es gibt zwei verschiedene Arten von Rahmenwerken, die sich in ihrer Verwendung unterscheiden (vgl. [JF88]):

- *Black-box-Rahmenwerk*

Bei der Verwendung eines Black-box-Rahmenwerkes muß der Entwickler keine Kenntnis des internen Aufbaus besitzen. Er benutzt die im Rahmenwerk speziell zur Verwendung vorgesehenen Klassen, die sich oftmals für die Gegebenheiten des Anwendungskontextes konfigurieren lassen.

- *White-box-Rahmenwerk*

Bei der Verwendung eines White-box-Rahmenwerkes ist es hingegen erforderlich, daß der Entwickler Kenntnis der internen Architektur besitzt. Das Rahmenwerk ist so konzipiert, daß der Entwickler von speziell dafür vorgesehenen Klassen Unterklassen ableitet, die den Anforderungen des Anwendungskontextes entsprechen.

Zwischen den beiden Arten von Rahmenwerken läßt sich oftmals keine starre Trennung vornehmen. So kann es vorkommen, daß einem White-box-Rahmenwerk durch sukzessive Weiterentwicklung Standardlösungen hinzugefügt werden, die auch eine black-box-artige Verwendung gestatten. Umgekehrt bieten Black-box-Rahmenwerke oftmals auch Möglichkeiten zur Spezialisierung einzelner Klassen an.

Ich bin der Auffassung, daß es für das Postversandsystem am geeignetsten ist, wenn dessen Softwarearchitektur ein White-box-Rahmenwerk ist, das auch eine black-box-artige Verwendung erlaubt. So besteht für den Entwickler die größtmögliche Flexibilität bei der anwendungsspezifischen Anpassung, indem er von speziell dafür vorgesehenen Klassen Unterklassen ableitet. Wenn es jedoch der Anwendungskontext erlaubt, kann er auch auf die bereits vorhandenen Standardlösungen zurückgreifen.

#### 4.4.2 Erweiterbarkeit

Die Eigenschaft Erweiterbarkeit ist für das Postversandsystem von großer Bedeutung, damit es für zusätzliche Anforderungen ausgestaltet werden kann.

## **Erweiterbarkeit**

"Erweiterbarkeit ist die Leichtigkeit, Softwareprodukte an Änderungen der Spezifikation anzupassen." ([Mey97], S. 6, Übersetzung d. Verf.)

Oftmals wird bei der Entwicklung von Softwaresystemen nicht bedacht, welche Erweiterungen notwendig sein können bzw. es geben könnte. Ich sehe das darin begründet, daß man zwar eine Vorstellung davon hat, welchen Anwendungsbereich ein Softwaresystem abdecken soll, aber die Vielzahl der verschiedenen Gegebenheiten selbst von einem Experten nicht überblickt werden kann.

Ähnlich verhält es sich auch bei dem Postversandsystem. Ich habe zwar eine Vorstellung davon, wofür es eingesetzt werden kann. Doch ist es mir nicht möglich, alle Einsatzgebiete vorhersehen zu können. Aus diesem Grund muß das Postversandsystem erweiterbar sein.

Um ein Softwaresystem so zu entwickeln, daß es nachträglich erweiterbar ist, hat Meyer (vgl. [Mey97]) zwei Prinzipien vorgeschlagen.

### *1. Einfaches Design*

Dieses Prinzip erleichtert es den Entwicklern, die eine Erweiterung vornehmen wollen, das bestehende Softwaresystem zu verstehen und zu erweitern.

### *2. Dezentralisation*

Dieses Prinzip sieht vor, daß die verschiedenen Funktionalitäten nach Möglichkeit in autonomen Modulen gekapselt werden. Damit wird bezweckt, daß sich eine Erweiterung nur auf wenige Module, auswirkt. Es soll vermieden werden, daß eine Erweiterung eine abhängigkeitsbedingte Änderung einer Vielzahl von Modulen bewirkt.

## 4.4.3 Anpaßbarkeit

Ähnlich wie bei der Eigenschaft Erweiterbarkeit (siehe Abschnitt 4.4.2) geht es bei der Anpaßbarkeit darum, daß das Postversandsystem in verschiedenen Anwendungskontexten eingesetzt werden kann. Jedoch liegt hier der Schwerpunkt darauf, daß die verwendete Softwarearchitektur nachträglich an dafür vorgesehenen Stellen anwendungsspezifisch angepaßt werden kann. Demeyer et al. definieren Anpaßbarkeit als:

"[...] designing software architectures that are easily adapted to target requirements." ([DMN+97], S. 60)

## **Anpaßbarkeit<sup>11</sup>**

Anpaßbarkeit ist die Einfachheit, ein Softwareprodukt so zu Modifizieren, daß es in verschiedenen aber ähnlichen Anwendungskontexten verwendet werden kann.

Um Anpaßbarkeit in einem Softwaresystem zu erreichen, ist es erforderlich, die anwendungsspezifischen Funktionalitäten explizit in einem Softwareentwurf vorzusehen. Schmidt benennt diese expliziten Funktionalitäten als *Hot-Spot*:

"A variable aspect of an application domain is called a hot spot." ([Sch97], S. 48)

---

<sup>11</sup> Anpaßbarkeit wird in der englischsprachigen Literatur als "tailorability" bezeichnet.

Wenn ein Softwaresystem so entworfen ist, daß lediglich die Hot-Spots die anwendungsspezifischen Funktionalitäten enthalten, so müssen auch nur diese modifiziert werden, wenn der Anwendungskontext sich ändert. Codenie et al. stellen deshalb fest:

"The hot spots define where the framework can be customized [...]" ([CHS+97], S. 73)

Mit diesem Prinzip verknüpft sich die Vorstellung, daß:

"[...] a software architecture that can be turned into a custom application by simply filling in the hot spots." ([CHS+97], S. 73)

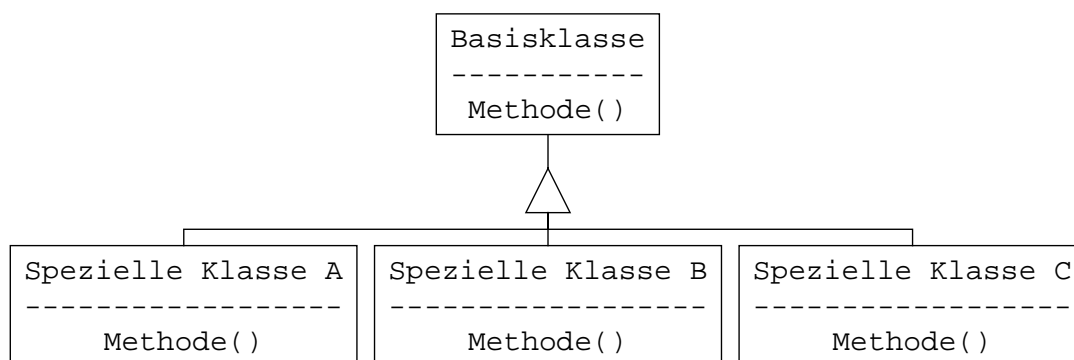
Auch Schmidt kommt zu einer ähnlichen Auffassung, indem er beschreibt, wozu Hot-Spots verwendet werden können:

"A hot spot lets you 'plug-in' an application-specific class or subsystem, either by selection from a set of those supplied with a black-box framework, or by programming a class or subsystem in a white-box framework." ([Sch97], S. 49)

Es stellt sich noch die Frage, wie ein Hot-Spot im Entwurf zu realisieren ist. Schmidt schlägt vor, daß ein Hot-Spot durch ein Subsystem implementiert wird. Für die Strukturierung eines Subsystems empfiehlt er die Verwendung von Entwurfsmustern:

"[...] a design pattern presents a proved solution for how to internally structure a hot spot subsystem in detail." ([Sch97], S. 50)

Im Regelfall ergibt sich dann eine Struktur, wie sie in Abbildung 12 dargestellt ist.



**Abbildung 12: Typische Struktur eines Hot-Spots**

In Abbildung 12 besteht die Struktur aus einer abstrakten Basisklasse und den davon abgeleiteten Klassen, die diese Schnittstelle in verschiedenen Variationen implementieren.

Bei dem Postversandssystem lassen sich verschiedene Hot-Spots identifizieren. So können z.B. die Materialien, die transportiert werden, in jedem Anwendungskontext sehr verschieden sein. Ein weiteres Beispiel sind die verschiedenen Posteingangsstrategien aus den Benutzungsmodellen im Abschnitt 4.3, die den Behältern verschiedene Eigenschaften als Posteingangskörbe verleihen. So kann der Benutzer im Benutzungsmodell mit dem erweiterten Umgang (siehe Abschnitt 4.3.2) beliebig viele Posteingangskörbe haben, wie z.B. einen Standardposteingangskorb und einen reservierbaren Posteingangskorb.

Im Abschnitt 5.3 werde ich bei den einzelnen Bestandteilen des Postversandsystems hervorheben, ob es sich um einen Hot-Spot handelt.

#### 4.4.4 Wiederverwendbarkeit

Die Eigenschaft Wiederverwendbarkeit soll dem Anspruch entsprechen, daß auch nur Teile des Postversandsystems in anderen Kontexten wiederverwendet werden können. Sie ist sehr eng mit der Kombinierbarkeit (siehe Abschnitt 4.4.5) verbunden.

##### **Wiederverwendbarkeit**

Die Wiederverwendbarkeit ist die Fähigkeit von Softwareelementen, für die Konstruktion von vielen verschiedenen Anwendungen zu dienen. ([Mey97], S. 7, Übersetzung d. Verf.)

Das Ziel ist, daß ähnliche Sachverhalte nicht wiederholt realisiert werden müssen. Das bedeutet aber auch, daß die wiederverwendbaren Softwareelemente eine besonders hohe Qualität und Robustheit aufweisen müssen. Die Forderung nach der hohen Qualität besteht deshalb, weil sonst u.a. ein Fehler ebenfalls "wiederverwendet" werden würde und somit eine größere Auswirkung hat, als wenn das Softwareelement nur an einer Stelle eingesetzt wird. Auch das Kriterium Robustheit spielt eine große Rolle, da durch die mehrfache Verwendung aus der Sicht des Softwareelements keine Annahme mehr über den Kontext und den Umgang getroffen werden kann.

Der erhöhte Aufwand für die Qualität und Robustheit kann aber mehr als kompensiert werden, weil das Softwareelement nur einmal entwickelt werden muß und nicht für jeden Einsatzkontext erneut.

Bei dem Postversandsystem bietet sich z.B. als wiederverwendbares Softwareelement ein Dienst an, der den reinen Transport von Materialien zwischen den Arbeitsplätzen erbringt. Dieses Softwareelement ließe sich z.B. in einem Terminplanungssystem, das auch die Verständigung über Konfliktsituationen unterstützt, wiederverwenden.

#### 4.4.5 Kombinierbarkeit

Die Eigenschaft Kombinierbarkeit bezieht sich auf die Wiederverwendbarkeit von Teilen des Postversandsystems, aber in einer noch flexibleren Art und Weise. Es geht dabei nicht mehr nur um die Wiederverwendung eines konkreten Softwareelementes. Es geht vielmehr darum, alle Softwareelemente, die unabhängig ihrer Implementierung die gleiche Schnittstelle aufweisen, in einem konkreten Kontext einsetzen zu können. Um diesen Unterschied zu verdeutlichen, spricht man nicht mehr von Softwareelementen sondern von Komponenten (vgl. [Sam97], [Szy98]).

Szyperski versteht unter einer Komponente:

"[...] a unit of composition with contractually specified interfaces and explicit context dependencies only. A software component can be deployed independently and is subject to composition by third parties." ([Szy98], S. 34)

Die Komponentenorientierung hat u.a. folgende Ziele:

1. Wiederverwendung einer Softwarekomponente in verschiedenen Softwaresystemen.
2. Verwendung verschieden implementierter Komponenten mit der gleichen Schnittstelle in einem konkreten Softwaresystem.



Die Eigenschaft Kombinierbarkeit setzt die Komponentenorientierung voraus, da Komponenten die notwendigen Eigenschaften aufweisen, um als Elemente eines Softwaresystems miteinander kombiniert werden zu können. Szyperski beschreibt diesen Zusammenhang folgendermaßen:

"[...] components are for composition. [...] Composition enables prefabricated 'things' to be reused by rearranging them in ever new composites." ([Szy98], S. 3)

### **Kombinierbarkeit**

Kombinierbarkeit ist die Fähigkeit von Softwarekomponenten, in verschiedenen Kombinationen wiederverwendet werden zu können.

Bei dem Postversandsystem ist die Kombinierbarkeit aus zwei Gründen von großer Bedeutung.

#### 1. *Wiederverwendbarkeit*

Es lassen sich aufgrund der Komponentenorientierung einzelne abgeschlossene Teile des Postversandsystems in anderen Anwendungskontexten wiederverwenden (siehe Abschnitt 4.4.4). So könnte z.B. der Teil des Postversandsystems, der den arbeitsplatzübergreifenden Transport von Materialien realisiert, auch in anderen Anwendungskontexten zum Zwecke des Materialaustauschs wiederverwendet werden.

#### 2. *Anpaßbarkeit*

Es können einzelne Teile des Postversandsystems zum Zwecke der Anpaßbarkeit (siehe Abschnitt 4.4.3) ausgetauscht werden. Beispielhaft dafür wäre, daß der von mir entwickelte Teil des Postversandsystems für den arbeitsplatzübergreifenden Transport von Materialien durch einen anderen ausgetauscht werden würde, weil sich die technischen Rahmenbedingungen für die notwendige Interprozeßkommunikation geändert haben.

Aus den genannten Gründen bin ich der Auffassung, daß die Softwarearchitektur des Postversandsystems zwar ein Rahmenwerk sein sollte, aber sich in Komponenten unterteilt.



## 5 Konzeption des Postversandsystems

In dem Kapitel 4 wurden die Anforderungen an das Postversandsystem aufgeführt. In diesem Kapitel konzipiere ich das Postversandsystem, so daß es den Anforderungen gerecht wird. Dabei leite ich zuerst im Abschnitt 5.1 den Entwurf insgesamt her und diskutiere, wie die insbesondere fachlichen Anforderungen erfüllt werden. Anschließend widme ich mich in Abschnitt 5.2 verschiedenen Fragestellungen, die sich bei der Konzeption gestellt haben, aber durch den Entwurf nicht beantwortet werden konnten. Im folgenden Abschnitt 5.3 werden die einzelnen Komponenten des Entwurfs im Detail vorgestellt. Dabei spreche ich verschiedene Entwurfsentscheidungen an und gehe insbesondere auf die technischen Anforderungen ein. Der abschließende Abschnitt 5.4 vermittelt einen Überblick über die Werkzeuge des Postversandsystems, da sie größtenteils die geforderten Benutzungsmodelle aus Abschnitt 4.3 verkörpern.

### 5.1 Entwurf

Ausgehend vom Leitbild vom Arbeitsplatz für eigenverantwortliche Expertentätigkeit (siehe Abschnitt 3.2) stellt sich die Frage, mit welcher Entwurfsmetapher dieses Leitbild in Bezug auf Unterstützung situativer kooperativer Arbeit ausgestaltet werden kann. Wirft man z.B. einen Blick in große Firmen, für die Arbeitsplätze für eigenverantwortliche Expertentätigkeit charakteristisch sind, so findet man oftmals zum Zwecke der expliziten Kooperation mittels Materialaustausch ein Postversandsystem in Form einer Hauspost vor. Deswegen verwende ich für den in diesem Abschnitt vorgestellten Entwurf des Postversandsystems die Hauspost als Entwurfsmetapher.

Der Entwurf orientiert sich sowohl im Aufbau als auch bei der Benennung der einzelnen Komponenten an der Entwurfsmetapher Hauspost. Insbesondere die Benennung habe ich bewußt so gewählt, da somit intuitiv schnell deutlich wird, was der Sinn und die Aufgaben der einzelnen Komponenten sind. Ich möchte an dieser Stelle betonen, daß hinter den realitätsnahen Begriffen oftmals sehr technische Konzepte stehen. Es ist nicht mein Ziel, mit den Komponenten die Vorbilder aus der Anwendungswelt nachzubauen.

Ausgehend von den Anforderungen werden im folgenden Probleme bei der Konzeption des Postversandsystems skizziert und zu deren Lösung einzelne Komponenten eingeführt. Anschließend wird erläutert, wie die Komponenten beim Versenden eines Materials zusammenspielen. Zum Schluß werde ich den Entwurf zusammenfassen und erklären, wie er die gestellten fachlichen Anforderungen erfüllt.

Der Ausgangspunkt ist, daß sich auf dem Arbeitsplatz des Benutzers Behälter für den Postein- und Postausgang befinden, die vom Postversandsystem bedient werden sollen. Hierbei meine ich den im Anwendungssystem modellierten Arbeitsplatz gemäß dem WAM-Ansatz. Auch in den folgenden Ausführungen beziehe ich mich immer auf den elektronischen Arbeitsplatz und nicht auf den realen in der Anwendungswelt.

Da das Postversandsystem Materialien zwischen verschiedenen Arbeitsplätzen transportiert, ist es mit dem folgenden Problem konfrontiert. Der Arbeitsplatz des Empfängers ist virtueller Natur und somit weder an einen physischen Ort gebunden noch jederzeit existent. Aus diesem

Grund kann nicht angenommen werden, ob und wo der Empfänger erreichbar ist. Trotzdem soll es in jedem Fall möglich sein, Materialien an den Empfänger zu senden.

Um dieses Problem zu lösen, habe ich die Postdienststelle eingeführt. Sie hat eine zentrale Stellung im Postversandsystem. Ihre Aufgabe ist es, ein zu transportierendes Material entgegenzunehmen, den Empfänger zu benachrichtigen und das Material zwischenzeitlich solange zu verwalten, bis es angefordert wird. Würde man statt dessen das zu transportierende Material direkt an den Empfänger übermitteln, so könnte es entweder verloren gehen oder der Versendevorgang wäre nicht durchführbar, wenn der Arbeitsplatz des Empfängers gerade nicht erreichbar ist.

Damit der Sinn und die Funktionsweise der Postdienststelle deutlicher werden, habe ich einen fachlich motivierten Aufbau gewählt, der sich aus einem Postbediensteten und einer Postfachwand zusammensetzt.

Die Postfachwand dient der Aufbewahrung und der Zuordnung der Materialien zu den Empfängeradressen. Sie ist ein interner Bestandteil der Postdienststelle und daher von außerhalb nicht zugreifbar. Sie besteht aus einer Vielzahl von Postfächern. Es gibt pro Empfängeradresse genau ein Postfach und umgekehrt. Damit ist eine umkehrbar eindeutige Zuordnung der Empfängeradressen zu Postfächern gewährleistet. Das macht nicht nur die Funktionsweise der Postdienststelle klarer, sondern vermeidet auch eine Mehrdeutigkeit, wenn zu einer Empfängeradresse die Materialien angefordert werden. Die Menge aller Postfächer ergeben die Menge aller gültigen Empfängeradressen.

Der Postbedienstete stellt die Schnittstelle der Postdienststelle nach außen hin dar. Er nimmt das zu transportierende Material entgegen und legt es in dem Postfach des Empfängers in der Postfachwand ab. Weiterhin ist es seine Aufgabe, zu versuchen, den Empfänger bezüglich des erhaltenen Materials zu benachrichtigen. Wenn sich der Empfänger auf diese Benachrichtigung meldet und das Material anfordert, entnimmt der Postbedienstete dieses aus der Postfachwand und händigt es aus.

Es kann vorkommen, daß es für den Empfänger kein Postfach in der Postfachwand gibt. In diesem Fall legt der Postbedienstete das Material in das Postfach des Absenders und benachrichtigt diesen.

Wie bereits erwähnt, transportiert das Postversandsystem Materialien zwischen verschiedenen Arbeitsplätzen. Zu diesem Zweck müssen die Materialien zuerst zur Postdienststelle gelangen und dann von dort aus zum Empfänger. Außerdem versendet die Postdienststelle Benachrichtigung an die Empfänger. Da jedoch die Arbeitsplätze der Benutzer sowie die Postdienststelle auf verschiedenen Computern angesiedelt sein können, ergibt sich das Problem, wie Informationen ausgetauscht werden können.

Um dieses Problem der sogenannten Interprozeßkommunikation zu lösen, sehe ich die spezielle Komponente Postzustelldienst vor. Sie kapselt die verwendete Middleware<sup>12</sup>. Es könnte z.B. sowohl RMI<sup>13</sup> als auch CORBA<sup>14</sup> eingesetzt werden. Welche Alternative die beste ist, kann nicht vorhergesehen werden und wird vom Anwendungskontext bestimmt. Aus

---

<sup>12</sup> Middleware ist Software zur Realisierung der Interprozeßkommunikation und dient somit dem Austausch von Informationen zwischen verschiedenen Computern.

<sup>13</sup> RMI steht für Remote Methode Invocation und ist eine Middleware in der Programmiersprache JAVA.

<sup>14</sup> CORBA steht für Common Object Request Broker Architecture und ist eine Middleware-Spezifikation, für die es von verschiedenen Herstellern Implementationen gibt.

diesem Grund bietet es sich auch an, eine Komponente für die Interprozeßkommunikation vorzusehen. Diese kann je nach Anwendungskontext verschieden implementiert werden.

Der Postzustelldienst dient dazu, die Materialien vom Arbeitsplatz des Benutzers zur Postdienststelle und zurück zu transportieren und die von der Postdienststelle veranlaßten Benachrichtigungen an alle Arbeitsplätze zu übermitteln, die sich für die entsprechende Empfängeradresse interessieren. Der Postzustelldienst setzt auf der Komponente Postdienststelle auf.

Eine fachliche Anforderung ist, daß mehrere Benutzungsmodelle möglich sind, da der jeweilige Anwendungskontext nicht vorhergesehen werden kann. Damit ergibt sich die Notwendigkeit, die Benutzungsmodelle flexibel zu halten. Das kann erreicht werden, indem eine Komponente explizit für die Benutzungsmodelle vorgesehen wird. Die Komponente Handhabung ist eine solche Komponente. Sie realisiert die in Abschnitt 4.3 beschriebenen Benutzungsmodelle. Es ist jedoch auch möglich, eine Handhabungskomponente zu bauen, welche die gleiche Schnittstelle hat, aber ein anderes Benutzungsmodell unterstützt.

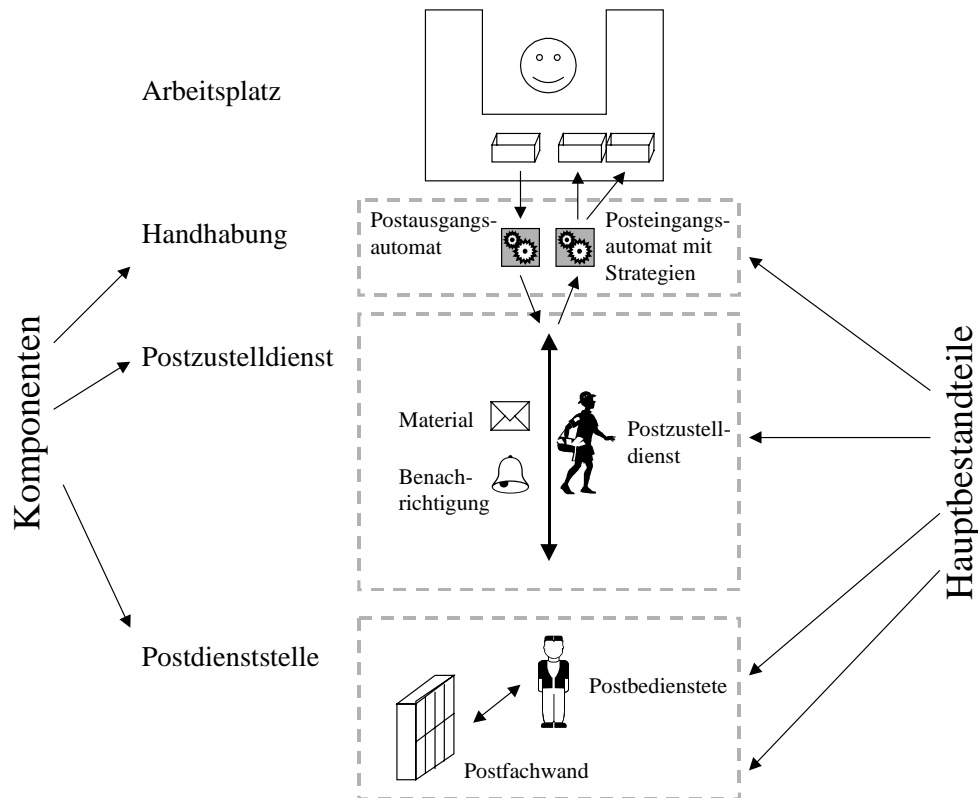
In den geforderten Benutzungsmodellen ist vorgesehen, daß der Benutzer einen bzw. beliebig viele Behälter als Posteingangskörbe nutzen kann. Von Bedeutung ist hierbei, daß je nach Konfiguration die Behälter nach einem bestimmten Kriterium vom Postversandsystem befüllt werden. Für die Lösung solcher Aufgaben bieten sich Automaten (siehe Abschnitt 3.3) an. So habe ich in der Handhabungskomponente einen Posteingangsautomaten für genau diese Aufgabe vorgesehen. Dieser Posteingangsautomat kann mit Posteingangsstrategien konfiguriert werden, die bestimmen, wie mit den empfangenen Materialien umzugehen ist. Gemäß dem erweiterten Umgang aus Abschnitt 4.3.2 gibt es zwei verschiedene Strategien, die Reservierungs- und die Standardstrategie. Die Reservierungsstrategie legt alle Materialien, die von einem bestimmten Absender empfangen wurden, in den ihr zugeordneten Behälter ab. Dagegen akzeptiert die Standardstrategie alle Materialien.

Die Funktionsweise des Posteingangsautomaten mit den Posteingangsstrategien ist wie folgt. Wird dem Posteingangsautomat eine Benachrichtigung vom Postzustelldienst über den Erhalt eines Materials in der Postdienststelle übermittelt, so befragt er all seine Posteingangsstrategien, ob sie diese Benachrichtigung bearbeiten können. Stimmt eine Strategie zu, so kann sie auf die Benachrichtigung angemessen reagieren. Die von mir vorgesehenen Posteingangsstrategien fordern das Material bei der Postdienststelle über den Postzustelldienst an und legen es anschließend in dem entsprechenden Posteingangskorb ab.

Neben den Posteingangskörben sind in den Benutzungsmodellen auch Behälter vorgesehen, die als Postausgangskörbe fungieren. Aus diesen Postausgangskörben sollen entweder automatisch oder manuell gestartet die Materialien versendet werden. Auch für diese Aufgabe bieten sich Automaten an. Zu diesem Zweck habe ich einen Postausgangsautomaten vorgesehen.

Der von mir vorgesehene Postausgangsautomat prüft in periodischen Zeitintervallen, ob sich Materialien in dem Postausgangskorb befinden. Ist das der Fall, so beauftragt er die Postdienststelle über den Postzustelldienst mit der Zustellung der Materialien an den Empfänger.

In Abbildung 13 ist der gesamte konzeptionelle Entwurf dargestellt. Es sind hier die drei Komponenten sichtbar, die abgrenzbare Teilaufgaben erfüllen und einander aufbauen.



**Abbildung 13: Konzeptioneller Entwurf des Postversandsystems**

In der beschriebenen Funktionsweise der einzelnen Komponenten ist aufgefallen, daß ein Material nicht direkt von einem Arbeitsplatz zum nächsten transportiert wird. Statt dessen wird es in der Postdienststelle abgegeben. Die Postdienststelle sendet es aber auch nicht unmittelbar an den Empfänger weiter, sondern versucht statt dessen, diesen zu benachrichtigen. Ein Grund für diese Vorgehensweise wurde bereits erwähnt. Es sind die virtuellen Arbeitsplätze, die keine Annahmen hinsichtlich deren Ort und Erreichbarkeit erlauben.

Ein weiterer Grund ist die Anforderung, daß prinzipiell beliebige Benutzungsmodelle möglich sein sollen. So könnte z.B. in einem Benutzungsmodell vorgesehen sein, daß ein in der Postdienststelle angekommenes Material vorerst auch dort verbleibt. Der Benutzer erhält jedoch einen Hinweis und sieht, was sich in dem Postfach befindet. Erst wenn er eigenverantwortlich dieses Material anfordert, um es auf seinem Arbeitsplatz zu bearbeiten, wird es ihm zugestellt. Dieses Vorgehen ist insbesondere bei Gruppenpostfächern vorteilhaft, bei denen mehrere Benutzer unter einer gemeinsamen Adresse erreichbar sind.

Wie die einzelnen vorgestellten Komponenten zusammenspielen, möchte ich nun anhand der Versendung eines Materials beschreiben.

1. Der Benutzer versieht das Material mit einer Empfängeradresse. Dazu verwendet er das Adressierungswerkzeug.
2. Der Benutzer legt das zu transportierende Material in den Postausgangskorb auf seinem Arbeitsplatz.

3. Der Postausgangsautomat der Komponente Handhabung entnimmt das zu transportierende Material aus dem Postausgangskorb und beauftragt den Postzustelldienst mit der Zustellung.
4. Der Postzustelldienst aus gleichnamiger Komponente transportiert das Material zur Postdienststelle und übergibt es dort dem Postbediensteten.
5. Der Postbedienstete der Postdienststelle nimmt das Material entgegen und legt es im Postfach mit der entsprechenden Empfängeradresse ab. Ist kein Postfach für diese Adresse vorhanden, so legt er es in das Postfach mit der Absenderadresse. Anschließend beauftragt er den Postzustelldienst mit der Benachrichtigung des Empfängers bzw. des Absenders.
6. Der Postzustelldienst transportiert die Benachrichtigung zu den entsprechenden Arbeitsplätzen und übergibt sie den Posteingangsautomaten aus der Handhabungskomponente.
7. Der Posteingangsautomat entscheidet nun anhand seiner Strategien, wie mit dieser Benachrichtigung zu verfahren ist und wird gewöhnlich den Postzustelldienst beauftragen, das Material zu liefern.
8. Der Postzustelldienst wendet sich nun an den Postbediensteten der Postdienststelle und fordert das Material anhand der Benachrichtigung an. Nachdem das Material ausgehändigt und zum Arbeitsplatz transportiert wurde, befindet es sich nicht mehr in der Postdienststelle. Es kann der Fall auftreten, wenn von mehreren Arbeitsplätzen aus das Material angefordert wird, daß der Postbedienstete dieses Material nicht aushändigen kann, da es bereits abgeholt wurde. In diesem Fall wäre das Zusammenspiel der Komponenten hiermit beendet.
9. Der Postzustelldienst übergibt dem Posteingangsautomaten das empfangene Material.
10. Der Posteingangsautomat entscheidet anhand seiner Strategien, wie mit dem erhaltenen Material zu verfahren ist und wird es gewöhnlich in einem Posteingangskorb ablegen.
11. Der Posteingangsautomat könnte nun z.B. noch eine visuelle Benachrichtigung des Benutzers veranlassen.

Der in diesem Abschnitt vorgestellte Entwurf erfüllt alle in Abschnitt 4 geforderten fachlichen Anforderungen, auf die ich nun im einzelnen eingehe.

#### 1. *Asynchrone verteilte Kooperation mit Hilfe von Kooperationsmitteln und -medien*

Es ist das Wesen des Postversandsystems, Materialien zwischen verschiedenen Arbeitsplätzen auszutauschen, d.h. die Ortsverschiedenheit der Kooperationspartner zu unterstützen. Da das Postversandsystem den Kooperationspartner erlaubt, zu einem beliebigen Zeitpunkt auf ein empfangenes Material zuzugreifen, wird die Zeitverschiedenheit der Kooperation unterstützt. Zusammenfassend läßt sich festhalten, daß das Postversandsystem asynchrone verteilte Kooperation unterstützt, da die Kooperationspartner sich zum Zwecke der Kooperation an verschiedenen Orten zu unterschiedlichen Zeiten aufhalten können.

Um die Kooperation zu unterstützen, werden Kooperationsmittel und -medien (siehe Abschnitt 2.2.3) angeboten. Das Kooperationsmedium, das den Austausch der Materialien realisiert, wird durch die Postkörbe vergegenständlicht. Als

Kooperationsmittel dienen die zu transportierenden Materialien, die Gegenstand der Kooperation sind.

2. *Verschiedenste Materialien können zwischen den Arbeitsplätzen ausgetauscht werden*

Im Entwurf wird keine Annahme darüber gemacht, welche Art von Materialien ausgetauscht werden können. Vielmehr wird davon ausgegangen, daß verschiedenste Materialien bei der Kooperation verwendet werden. In Abschnitt 5.2.2 gehe ich darauf ein, welche Anforderungen das Material erfüllen muß, damit es mit Hilfe des Postversandsystems transportiert werden kann.

3. *Einheit von Ort und Zeit*

Die Voraussetzung dafür, daß die Einheit von Ort und Zeit der Materialien sichergestellt werden kann ist, daß ein Material immer nur an eine Empfängeradresse mit dem Postversandsystem zeitgleich gesendet werden kann. Ansonsten sorgt das Postversandsystem dafür, daß sich ein Material beim Versenden zu einem Zeitpunkt entweder noch beim Sender oder in der Postdienststelle oder beim Empfänger befindet.

4. *Unterstützung verschiedener Benutzungsmodelle*

Dieser fachlichen Anforderung wird Rechnung getragen, indem die spezielle austauschbare Handhabungskomponente für die Benutzungsmodelle vorgesehen ist. Diese Komponente kann verschiedenartig implementiert werden, so daß nahezu beliebige Benutzungsmodelle realisierbar sind. Ich möchte jedoch darauf hinweisen, daß die durch die Handhabungskomponente möglichen Benutzungsmodelle immer nur im Bereich der asynchronen verteilten Kooperation liegen können. Wie die in Abschnitt 4.3 geforderten Benutzungsmodelle umgesetzt worden sind, ist in Abschnitt 5.3.1 beschrieben.

Die von den Komponenten erbrachten Dienstleistungen lassen sich mit den in Kapitel 2 eingeführten Klassifizierungen wie folgt einordnen.

1. Alle Komponenten des Postversandsystems sind von ihrer Funktionalität her so konzipiert, daß sich die Benutzer zu verschiedenen Zeiten an unterschiedlichen Orten befinden können, um Materialien auszutauschen. Deshalb läßt sich die zusammen erbrachte Dienstleistung der *asynchronen verteilten Interaktion* zuordnen (siehe Abschnitt 2.3).
2. Zusammen mit der Postdienststelle ermöglicht der Postzustelldienst den arbeitsplatzübergreifenden Austausch von Materialien. Diese Dienstleistung entspricht einer *Kommunikationsunterstützung* (siehe Abschnitt 2.2.1).
3. Die Handhabungskomponente setzt auf die Dienste der anderen beiden Komponenten auf. Sie realisiert einen anwendungsfachlich motivierten Umgang mit dem Postversandsystem, wie das Befüllen der Posteingangskörbe nach bestimmten Kriterien. Zu diesem Zweck stellt sie Kooperationsmittel und -medien zur Verfügung, mit denen der Arbeitsplatz des Benutzers ausgestaltet werden kann. Diese orientieren sich an Entwurfsmetaphern der Anwendungswelt und erlauben ein situatives kooperatives Arbeitshandeln. Die von der Handhabungskomponente erbrachte Dienstleistung entspricht somit der *Kooperationsunterstützung* (siehe Abschnitt 2.2.3).



## **5.2 Fragestellungen bei der Konzeption**

Bei der Konzeption des Postversandsystems haben sich mir noch weitere Fragestellungen ergeben, die sich nicht mit dem Entwurf beantworten ließen. Auf diese Fragestellungen gehe ich in den folgenden Abschnitten ein. Als Antwort führe ich meistens spezielle Schnittstellen ein. Diese Schnittstellen lassen sich nicht einer konkreten Komponente zuordnen, weshalb ich sie auch in diesem Abschnitt diskutiere. Jedoch werden sie bei der Konzeption der einzelnen im Entwurf beschriebenen Komponenten (siehe Abschnitt 5.3) benötigt.

### **5.2.1 Adressierung**

Damit ein Material eindeutig einem Empfänger zugeordnet werden kann, muß es mit einer entsprechenden Empfängeradresse versehen werden. Dieser einfache Sachverhalt birgt noch eine Reihe weiterer Fragen. So ist z.B. noch keine Aussage darüber getroffen, welche Eigenschaften eine Adresse haben sollte und wie sie im Postversandsystem realisiert ist. Weiterhin ist offen, ob sich eine Adresse eindeutig auf einen Arbeitsplatz bezieht. Aufgrund all der aufgeworfenen Fragen, greife ich die Problematik der Adressierung in diesem Abschnitt auf.

#### **Wie kann der Empfänger identifiziert werden?**

Damit ein Material auch beim gewünschten Empfänger ankommt, muß dieser identifizierbar sein. Da ich mich an der Entwurfsmetapher Hauspost orientiere, erscheint es nur allzu offensichtlich, daß für diese Problematik Adressen verwendet werden.

Zu diesem Zweck führe ich die Klasse *Transportadresse* ein. Sie stellt einen Fachwert (vgl. [Zül98], S. 316ff.) dar und besitzt deswegen eine Wertsemantik. Diese Entwurfsentscheidung ist sowohl fachlich als auch technisch motiviert. Der fachliche Grund ist, daß die *Transportadresse* der Identifizierung dient und daher der Wert von Interesse ist und nicht das Objekt selbst, das den Wert verkörpert. Aus technischer Sicht wird der Umgang wesentlich vereinfacht, da Seiteneffekte vermieden werden, wie sie bei der Referenzsemantik auftreten können.

Um der Anforderung nach Anpaßbarkeit (siehe Abschnitt 4.4.3) gerecht zu werden, habe ich die Klasse *Transportadresse* als einen Hot-Spot vorgesehen, da sie die Entwurfsentscheidung verkörpert, wie etwas adressiert wird. Sie stellt eine Schnittstelle zur Verfügung, die eine konkrete Implementation erfüllen muß. So bleibt verborgen, wie sie implementiert ist.

Insbesondere bei der Implementation sind verschiedene Varianten möglich. Ich habe mich für eine sehr einfache entschieden, die den Anforderungen aus Abschnitt 4 genügt. Diese sieht vor, daß die *Transportadresse* zu einem im Anwendungssystem modellierten Benutzer korreliert. Eine mögliche Erweiterung wäre ein hierarchischer Adreßraum.

Da die Klasse *Transportadresse* verschiedenartig realisiert werden kann, stellt die Erzeugung ein weiteres Problem dar. Ich sehe die Zuständigkeit für die Erzeugung nicht bei dem Postversandsystem angesiedelt. Da von mir die Entwurfsentscheidung gefällt wurde, daß eine Adresse zu einem Benutzer korreliert, habe ich die Erzeugungsmethode an der Klasse selbst vorgesehen (siehe Abbildung 14), die mit einem Benutzerobjekt parametrisiert wird.

Je nach Anwendungskontext kann es notwendig werden, daß ein separater Dienst Adressen erzeugt. Das ist insbesondere dann der Fall, wenn sie in verschiedenen Anwendungssystemen, wie z.B. bei der Registratur und dem Vorgangsmontor, verwendet werden.

```
Transportadresse
-----
erzeuge(Benutzer)
als Text ()
ist gleich (Transportadresse)
```

#### **Abbildung 14: Umgangsformen der Klasse Transportadresse**

Eine weitere Umgangsform der Klasse ist, daß sie eine Textrepräsentation von ihrem Wert liefern kann. Dieser Text wird insbesondere zur visuellen Darstellung einer Adresse benötigt.

Damit die `Transportadresse` gemäß ihrer eigentlichen Bestimmung zur Identifikation verwendet werden kann, hat sie eine Funktion, um auf Wertgleichheit mit anderen Exemplaren geprüft zu werden.

#### **Sollte sich die Adresse eindeutig auf einen Arbeitsplatz beziehen?**

Eine eindeutige Abbildung einer Empfängeradresse auf die Arbeitsplätze muß nicht gegeben sein. Der Entwurf sieht zwar eine eindeutige Zuordnung der Empfängeradressen zu Postfächern vor, jedoch wird nicht vorausgesetzt, daß die Empfängeradressen auch eindeutig für die Arbeitsplätze sind. Es ist durchaus vorstellbar, daß mehrere Postfächer für einen Arbeitsplatz bestimmt sind bzw. mehrere Arbeitsplätze sich ein Postfach teilen. Dadurch wird die Möglichkeit geschaffen, von einer arbeitsplatzorientierten Adressierung abzuweichen. So wäre es z.B. denkbar, daß neben den Arbeitsplätzen auch Rollen von Personen, die sie in Organisationen einnehmen können, adressiert werden. Ich möchte an dieser Stelle nicht auf ein Rollenkonzept und dessen Computerunterstützung eingehen, sondern den Vorteil der genannten Möglichkeit lediglich am folgenden Beispiel verdeutlichen:

Als Mitarbeiter einer großen Softwarefirma arbeite ich an meinem Arbeitsplatz mit einem Computer. Wenn ich nun ein Problem mit dem Computer habe, sende ich eine elektronische Nachricht an den Support. Dabei ist es für mich völlig unerheblich, welche Personen in der Firma für den Support zuständig sind. Herr Meyer ist ein für den Support zuständiger Mitarbeiter. Er hat auf seinem Arbeitsplatz zwei Posteingangskörbe. Der erste ist für die an ihn persönlich gerichtete Post und der zweite für die Nachrichten an den Support. Da meine Nachricht in seinem Posteingangskorb für den Support abgelegt wurde, wird er sich meinem Problem annehmen.

Das Beispiel demonstriert gleich beide oben genannten Fälle hinsichtlich der Zuordnung von Postfächern zu Arbeitsplätzen.

1. Die Postkörbe auf dem Arbeitsplatz von Herrn Meyer werden sowohl mit der Post aus seinem persönlichen Postfach befüllt als auch mit den Nachrichten aus dem Postfach für den Support. Dem Arbeitsplatz von Herrn Meyer können also zwei Postfächer aus der Postdienststelle zugeordnet werden.
2. Da Herr Meyer nicht die einzige für den Support zuständige Person in der Firma ist, können auch andere Mitarbeiter mit den Nachrichten aus dem Postfach für den Support

versorgt werden, d.h. sie teilen sich ein Postfach. Aus dieser Konstellation ergibt sich zwangsläufig eine Konkurrenzsituation in bezug auf die Zuordnung einer Nachricht für den Support zu den möglichen Arbeitsplätzen. Um diese Konkurrenzsituation aufzulösen, bedarf es eines zusätzlichen Aufwandes. Beispielsweise wäre es vorstellbar, daß die Handhabungskomponente für die Unterstützung solcher Fälle in der Art angepaßt wird, daß jeder Supportmitarbeiter erst eine Mitteilung darüber erhält, daß sich eine Nachricht in dem gemeinsamen Postfach befindet. Erst wenn der Supportmitarbeiter die Nachricht explizit anfordert, wird sie in dem Posteingangskorb auf seinem Arbeitsplatz abgelegt.

Da ich in den Benutzungsmodellen (siehe Abschnitt 4.3) diese Möglichkeit der Zuordnung von Postfächern zu Arbeitsplätzen nicht vorgesehen habe, werde ich auch nicht im weiteren darauf eingehen.

## 5.2.2 Transportierbarkeit von Materialien

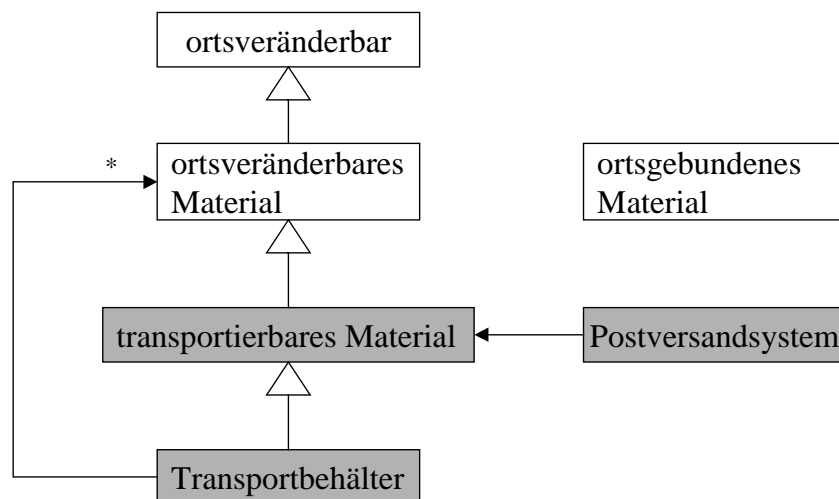
Es gibt eine Reihe von Problemen, die sich auf den Transport von Materialien beziehen. So stellt sich z.B. die Frage, welche Voraussetzungen ein Material erfüllen muß, damit es mit dem Postversandsystem transportiert werden kann und welche Möglichkeiten es gibt, ein Material transportierbar zu machen. Neben diesen grundsätzlichen Fragen gibt es auch noch solche, die weiterführende Probleme ansprechen. Deswegen diskutiere ich in diesem Abschnitt all die Fragen, die sich um die Problematik der Transportierbarkeit von Materialien ranken.

### **Wie wird ein Material transportierbar?**

Die Hauptaufgabe des Postversandsystems ist es, Materialien zwischen verschiedenen Arbeitsplätzen zu transportieren. Nun ist es offensichtlich, daß nicht jedes beliebige Material transportiert werden kann, sondern es müssen gewisse Voraussetzungen erfüllt sein. In Anlehnung an die Entwurfsmetapher Hauspost sind die Voraussetzungen, daß das Material seinen Ort wechseln kann und notwendige Transportinformationen anbietet. Ein Material, das diese Voraussetzungen erfüllt, nenne ich ein *transportierbares Material*.

In der Abbildung 15 ist dargestellt, wie ich ein transportierbares Material im Verhältnis zu anderen Materialien aus der Sicht des Postversandsystems einordne.

Damit ein Material seinen Ort wechseln kann, muß es *ortsveränderbar* sein. Diese Eigenschaft ist sowohl fachlich als auch technisch relevant. Aus fachlicher Sicht sind die meisten Materialien ortsveränderbar, die einen Arbeitskontext ausgestalten, jedoch kann es einzelne geben, die z.B. aufgrund ihrer Komplexität oder aus juristischen Gründen an einen Ort gebunden sind. Damit sichergestellt wird, daß nur die gewünschten Materialien den Ort wechseln können, müssen sie die Eigenschaft der Ortsveränderbarkeit aufweisen. Aus technischer Sicht können nur solche Objekte zwischen verschiedenen Prozeßräumen übertragen werden, die sich in einen Datenstrom umwandeln lassen. Dieser Vorgang der Umwandlung wird als Serialisierung bezeichnet. Somit können nur solche Objekte ausgetauscht werden, die serialisierbar sind.



**Abbildung 15: Einordnung eines transportierbaren Materials aus der Sicht des Postversandsystems**

Eine weitere notwendige Eigenschaft eines Materials ist, damit es mit dem Postversandsystem versendet werden kann, daß es *Transportinformationen* anbietet. Diese Transportinformationen setzen sich zusammen aus der Empfängeradresse, dem Absender, dem Zustelldatum und einer Materialidentifikation.

Aus der Kombination der beiden genannten Eigenschaften ergeben sich aus der Sicht des Postversandsystems die folgenden drei Klassen von Materialien:

1. *ortsgebundene Materialien*

Diese sind für das Postversandsystem ohne weitere Bedeutung, da sie ihren Ort nicht wechseln können.

2. *transportierbare Materialien*

Sie haben beide Eigenschaften und sind somit unmittelbar durch das Postversandsystem versendbar.

3. *ortsveränderbare Materialien*

Diese Materialien sind durch das Postversandsystem nicht unmittelbar transportierbar, da sie keine Transportinformationen anbieten. So bieten auch die Materialien aus der Anwendungswelt nur in den wenigsten Fällen Transportinformationen an. Vielmehr bedient man sich sogenannter Transportbehälter, wie z.B. einer Versandkiste oder einem Briefumschlag, in welche die ortsveränderbaren Materialien abgelegt werden. Diese Transportbehälter umschließen die Materialien und bieten dem Postversandsystem die notwendigen Informationen an. Abschnitt 5.3.1.5 stellt einen solchen Transportbehälter vor, wie er vom Postversandsystem zur Verfügung gestellt wird.

Bei der Realisierung des Postversandsystems konnte ich bei den in der Abbildung 15 dargestellten Schnittstellen zum Teil auf bereits vorhandene zurückgreifen. So bietet die Programmiersprache JAVA für die Eigenschaft "ortsveränderbar" die Schnittstelle `Serializable` an, während für die Klasse der ortsveränderbaren Materialien das JWAM-Rahmenwerk die Schnittstelle `Thing` (vgl. [Lip99], S. 77f.) definiert. Im übrigen wird diese

Schnittstelle von allen mit dem JWAM-Rahmenwerk erstellten Materialien erfüllt, so daß sie alle mit dem Postversandsystem versendet werden können (siehe unten).

Wie aus der Abbildung 15 hervorgeht, kann das Postversandsystem nur mit transportierbaren Materialien unmittelbar umgehen. Es stellt sich nun die Frage, was zu tun ist, damit ein Material mit dem Postversandsystem transportiert werden kann. Prinzipiell gibt es folgende zwei Ansätze.

1. *Das Material ist ein transportierbares Material.*

Dieser Ansatz kann nur bei der Entwicklung eines Materials angewendet werden, denn er bedeutet, daß das Material alle Eigenschaften eines transportierbaren Materials erben muß. Diese Vorgehensweise ist insbesondere dann angebracht, wenn ein Material speziell für den Transport mit dem Postversandsystem gedacht ist, wie z.B. Vorgangsmappen.

2. *Das Material ist ortsveränderbar und wird in einem Transportbehälter abgelegt.*

Meiner Meinung nach dürfte dieser Ansatz der gewöhnliche Weg sein, denn die meisten Materialien werden unabhängig von dem Postversandsystem entwickelt. Es gibt natürlich Materialien, die für den Transport mit dem Postversandsystem gedacht sind, wie z.B. Vorgangsmappen, jedoch ist der Zweck der meisten Materialien unabhängig davon. Nun wäre das Ziel des Postversandsystems aber weit verfehlt, wenn nicht auch die Materialien versendet werden könnten, die nicht vordergründig für den Transport mit dem Postversandsystem ausgelegt sind. Aus diesem Grund besteht die Möglichkeit, ein ortsveränderbares Material in einem Transportbehälter zu verpacken, der wiederum transportierbar ist. An dieser Stelle möchte ich nochmals darauf hinweisen, daß alle Materialien, die mit dem JWAM-Rahmenwerk entwickelt werden, ortsveränderbare Materialien sind. Somit können alle Materialien, die mit dem JWAM-Rahmenwerk entwickelt wurden, mit dem Postversandsystem transportiert werden.

Leider kehrt sich der Vorteil, daß der Transportbehälter das Material kapselt, beim Empfänger in einen Nachteil um. Dieser kann nämlich auch nur erkennen, daß er einen Transportbehälter empfangen hat. Um zu erkennen, was der eigentliche Inhalt der Sendung war, muß er den Transportbehälter erst öffnen.

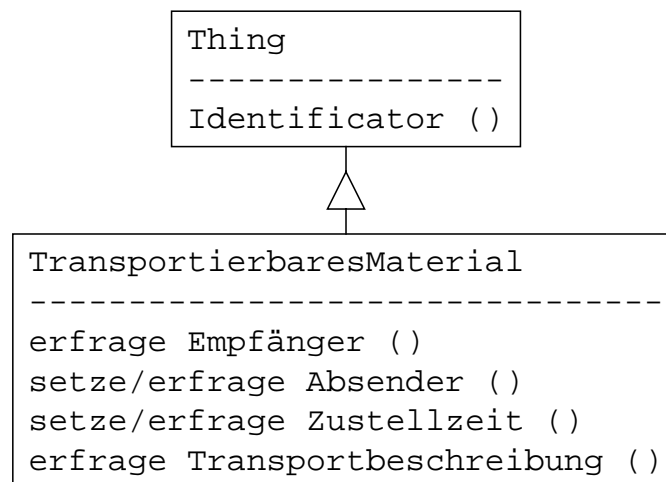
### **Wie wird die Menge der transportierbaren Materialien definiert?**

Um die Menge der transportierbaren Materialien im Postversandsystem definieren zu können, führe ich den gleichnamigen Aspekt ein. Dieser Aspekt `TransportierbaresMaterial` verkörpert alle notwendigen Informationen sowie Eigenschaften eines Materials, damit es mit Hilfe des Postversandsystems unmittelbar versendet werden kann. Das bedeutet konkret, daß nur ein Material, das diesen Aspekt besitzt, mit dem Postversandsystem transportiert wird.

Gemäß der Forderung nach Basisfunktionalität ist der Aspekt `TransportierbaresMaterial` als ein Hot-Spot vorgesehen (siehe Abschnitt 4.4.3). Er faßt alle für das Postversandsystem notwendigen Eigenschaften der zu transportierenden Materialien zusammen und abstrahiert aus der Sicht des Postversandsystems von der konkreten Implementation. Dadurch können nachträglich beliebige anwendungsspezifische Materialien entwickelt werden, die mit dem Postversandsystem versendet werden können, wenn sie diesen Aspekt erfüllen. Ein Beispiel dafür ist die Vorgangsmappe (vgl. [Bre00]).

Der Aspekt `TransportierbaresMaterial` erweitert die Schnittstelle `Thing` des JWAM-Rahmenwerkes. Somit ist gewährleistet, daß ein Material, das diesen Aspekt erfüllt, auch alle Eigenschaften eines Materials im Sinne des JWAM-Rahmenwerkes hat.

Die Umgangsformen des Aspektes `TransportierbaresMaterial` orientieren sich an den gestellten Anforderungen hinsichtlich der Bereitstellung von Transportinformationen (siehe Abbildung 16).



**Abbildung 16: Umgangsformen des Aspektes `TransportierbaresMaterial`**

So kann die Empfängeradresse erfragt werden, was für das Postversandsystem von Wichtigkeit ist. Es ist nicht vorgesehen, daß die Empfängeradresse auch gesetzt werden kann, da diese Funktionalität für den Transport des Materials aus der Sicht des Postversandsystems irrelevant ist. Diese Funktionalität trotzdem mit vorzusehen, kann wegen dem Anspruch auf Basisfunktionalität zu Problemen führen. So kann es durchaus Anwendungsfälle geben, in denen am Material selbst keine Empfängeradresse gesetzt werden darf. Ein Beispiel hierfür ist die Vorgangsmappe mit Laufzettel von Breitling (vgl. [Bre00]). In diesem Fall legt der Laufzettel fest, wer der Empfänger ist. Wenn es möglich wäre, über eine Methode des Aspektes `TransportierbaresMaterial` die Empfängeradresse mit Hilfe des Adressierungswerkzeuges zu setzen, würde das bedeuten, daß der Eintrag im Laufzettel geändert wird. Das jedoch ist gewiß nicht gewollt.

Weiterhin kann der Absender gesetzt und erfragt werden. Das Setzen des Absenders wird vom Postausgangsautomaten vorgenommen, wenn etwas versendet wird. Das Erfragen des Absenders dient dem Postbediensteten in dem Fall, daß die Empfängeradresse falsch ist und das Material in das Postfach des Absenders zurückgelegt werden soll.

Rein informellen Charakter hat die Zustellzeit. Diese wird ebenfalls automatisch vom Postversandsystem gesetzt und kann von anderen Anwendungen ausgewertet werden. Sie hat für das Postversandsystem keine weitere Bedeutung.

Da der Aspekt `TransportierbaresMaterial` die Schnittstelle `Thing` erweitert, kann ein Identifikator des Materials erfragt werden. Dieser identifiziert das Material eindeutig im gesamten Anwendungssystem. Der Identifikator wird im Postversandsystem benötigt, wenn der Empfänger ein Material aus seinem Postfach von der Postdienststelle anfordert.

Ebenfalls sehr wichtig für das Postversandsystem ist die Transportbeschreibung, die an dem Material erfragt werden kann. Diese wird mit der folgenden Fragestellung diskutiert.

### **Ist eine Beschreibung des Materials vorteilhaft und wie kann sie realisiert werden?**

Im Entwurf (siehe Abschnitt 5.1) ist aufgrund der virtuellen Natur der modellierten Arbeitsplätze nicht vorgesehen, daß die Materialien direkt zwischen den Arbeitsplätzen transportiert werden. Vielmehr wird das zu transportierende Material der Postdienststelle übergeben, welche daraufhin den Empfänger über den Erhalt benachrichtigt. Erreicht die Benachrichtigung die Handhabungskomponente des Empfängers, so wird sie das Material bei der Postdienststelle anfordern und anschließend in einen Posteingangskorb auf dem Arbeitsplatz ablegen.

Damit die Benachrichtigung ihren Zweck auch erfüllt, muß sie alle notwendigen Informationen über das Material enthalten. Prinzipiell wäre es wahrscheinlich am einfachsten, wenn die Benachrichtigung eine Kopie des Materials enthält. Dann könnten auf der Empfängerseite an diesem Exemplar alle Informationen direkt erfragt werden. Leider besteht aber das Problem, daß die Materialien sehr umfangreich sein können. Deshalb könnte die Benachrichtigung mitsamt der Kopie des Materials zu einer unnötigen Belastung der Systemressourcen führen. Ich denke insbesondere an den Netzwerkverkehr, der damit gesteigert wird.

Zur Lösung des Problems sehe ich vor, daß an dem transportierbaren Material (siehe Abschnitt 5.2.2) eine Transportbeschreibung erfragt werden kann. Sie enthält alle notwendigen Informationen, um das entsprechende Material ausreichend zu beschreiben. Insbesondere beziehe ich mich auf die für das Postversandsystem wichtigen Transportinformationen, wie z.B. die Empfänger- und die Absenderadresse. Damit eignet sich die Transportbeschreibung für die bereits erwähnte Benachrichtigung.

Da die Transportbeschreibung lediglich die Transportinformationen enthält, ist sie wesentlich leichtgewichtiger in bezug auf den Kapazitätsbedarf als das beschriebene Material. Diese Eigenschaft wirkt sich in folgenden Situationen vorteilhaft aus, in denen sie im Postversandsystem benötigt wird und es zu einer Netzwerkkommunikation kommt:

#### *1. Benachrichtigung des Empfängers durch die Postdienststelle*

Die Postdienststelle (siehe Abschnitt 5.3.3) benachrichtigt den Empfänger über den Neuzugang in seinem Postfach, indem sie ihm die Transportbeschreibung des empfangenen Materials zusendet. Der Empfänger erhält damit alle notwendigen Informationen, um zu entscheiden, ob und wie er auf den Neuzugang zu reagieren hat.

#### *2. Erstellung des Inhaltsverzeichnisses eines Postfaches*

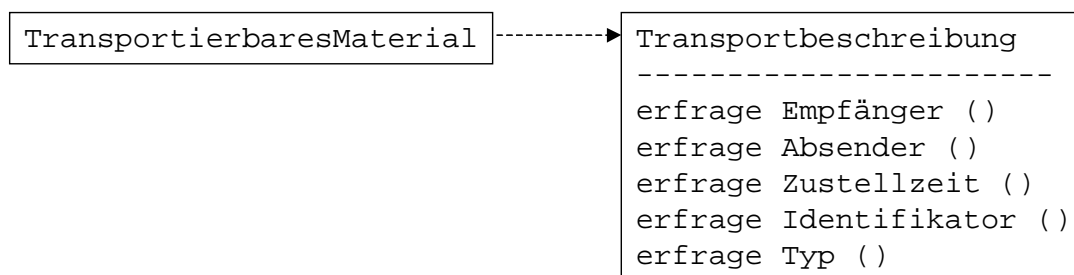
Der Inhaber eines Postfaches kann über den Postzustelldienst bei der Postdienststelle das Inhaltsverzeichnis seines Postfaches erfragen. Dieses Inhaltsverzeichnis wird durch die Postdienststelle erzeugt und setzt sich aus den Transportbeschreibungen der Materialien in dem Postfach zusammen. Anhand dieses Inhaltsverzeichnisses weiß der Benutzer, was sich in seinem Postfach befindet und kann die einzelnen Materialien anfordern.

### 3. Anforderung eines empfangenen Materials

Wenn z.B. der Empfänger durch die Postdienststelle über den Erhalt eines Materials benachrichtigt wurde, so kann er es mit der erhaltenen Transportbeschreibung über den Postzustelldienst (siehe Abschnitt 5.3.2) bei der Postdienststelle anfordern. Diese Transportbeschreibung enthält alle notwendigen Informationen, damit der Postbedienstete das angeforderte Material in der Postfachwand auffinden und zur Verfügung stellen kann.

Zur Realisierung der Transportbeschreibung führe ich die gleichnamige Klasse ein. Sie ist als ein Fachwert realisiert und hat somit Wertsemantik. Diese Entwurfsentscheidung läßt sich sowohl fachlich als auch technisch begründen. Fachlich ist ein Exemplar der Transportbeschreibung ein Wert, der das zu transportierende Material näher beschreibt. Dabei steht der Wert im Vordergrund und nicht das Objekt. Das bedeutet aber auch, daß wenn sich der Zustand des Materials ändert, indem z.B. eine neue Empfängeradresse gesetzt wird, eine zuvor gelieferte Transportbeschreibung unverändert bleibt. Erst eine neu vom Material erfragte Transportbeschreibung kann diese für den Transport relevante Änderung enthalten. Diese Verhaltensweise ist aber für das Postversandsystem unkritisch, da der Zustand des zu transportierenden Materials während des gesamten Transportvorganges nicht geändert wird und somit der Wert einer konkreten Transportbeschreibung für das Material signifikant bleibt. Aus technischer Sicht läßt sich mit einem Wert einfacher umgehen, da evtl. Seiteneffekte ausgeschlossen sind.

Eine Anforderung an das Postversandsystem ist, daß verschiedenste Materialien transportiert werden können. Daraus resultiert, daß auch deren Beschreibung verschieden ausfallen kann. Um Anpassung zu ermöglichen, stellt die Klasse Transportbeschreibung einen Hot-Spot (siehe Abschnitt 4.4.3) dar. Sie enthält all die für den Transport notwendigen Informationen, wie z.B. die Empfänger- und die Absenderadresse.



**Abbildung 17: Umgangsformen der Klasse Transportbeschreibung**

In Abbildung 17 sind die Umgangsformen der Transportbeschreibung dargestellt. Sie bestehen lediglich aus sondierenden Operationen für die Transportinformationen. Es können somit keine neuen Werte gesetzt werden, was auch dem Charakter eines Fachwertes widersprechen würde.

Die in Abbildung 17 dargestellten Umgangsformen entsprechen fast vollständig denen des Aspektes TransportierbaresMaterial (siehe Abschnitt 5.2.2). Der Grund dafür ist, daß die Transportbeschreibung ein komplexer Fachwert ist, der sich aus mehreren Fachwerten, wie z.B. den Transportadressen, zusammensetzt. Da diese einzelnen Fachwerte erfragbar sein sollen, ergeben sich zwangsläufig die gleichen Umgangsformen.



Gegenüber dem Aspekt `TransportierbaresMaterial` gibt es eine neue Umgangsform. Mit ihr kann der Typ des Materials erfragt werden. Diese Umgangsform erlaubt der Handhabungskomponente in Abhängigkeit des Typs auf eine Benachrichtigung zu reagieren.

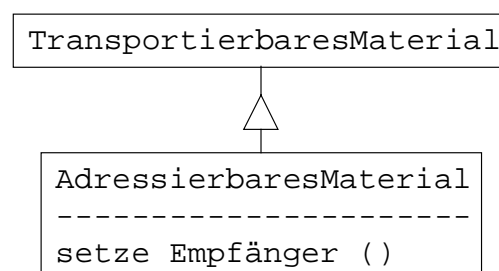
Aufgrund der redundanten Umgangsformen der Transportbeschreibung gegenüber dem Aspekt `TransportierbaresMaterial` drängt sich die Frage auf, warum nicht alle Transportinformationen bei der Transportbeschreibung angesiedelt werden, anstatt sie bei dem transportierbaren Material vorzusehen. Schließlich könnten so die redundanten Umgangsformen gespart werden. Diese Entwurfsentscheidung würde aber bedeuten, daß die Transportbeschreibung kein Fachwert mehr sein kann. Damit würden auch alle Vorteile, die sich daraus ergeben (siehe oben), verloren gehen. Aus diesem Grund halte ich die von mir beschriebene Entwurfsentscheidung für angebrachter.

### Wie kann für ein Material festgelegt werden, wohin es versendet werden soll?

In den geforderten Benutzungsmustern ist vorgesehen, daß der Benutzer mit Hilfe eines Adressierungswerkzeuges die Empfängeradresse des zu transportierenden Materials setzen kann. Allerdings bietet der Aspekt `TransportierbaresMaterial` keine entsprechende Umgangsform dafür an.

Aus diesem Grund führe ich den Aspekt `AdressierbaresMaterial` ein. Er ist eine Erweiterung des Aspektes `TransportierbaresMaterial` und erlaubt das Setzen der Empfängeradresse (siehe Abbildung 18).

Der Aspekt `AdressierbaresMaterial` ist nur für das Adressierungswerkzeug (siehe Abschnitt 5.4.1) von Bedeutung, welches das Setzen der Empfängeradresse für ein zu transportierendes Material unterstützt. Ansonsten hat der Aspekt für das Postversandsystem keine weitere Funktion.

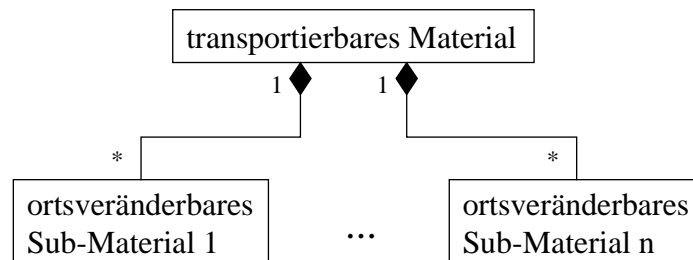


**Abbildung 18: Umgangsformen des Aspektes `AdressierbaresMaterial`**

In Abbildung 18 wird deutlich, daß der Aspekt `AdressierbaresMaterial` vom Aspekt `TransportierbaresMaterial` erbt und um die Umgangsform zum Setzen der Empfängeradresse erweitert. Aufgrund der Vererbung ist ein adressierbares Material auch transportierbar.

## Eignen sich alle Materialien für den Transport?

Aus technischer Sicht könnten theoretisch alle Materialien mit dem Postversandsystem transportiert werden, die den Aspekt `TransportierbaresMaterial` erfüllen. Allerdings beschränke ich meine Betrachtungen bei den zu transportierenden Materialien auf die mit der in Abbildung 19 dargestellten Struktur und einem handlichen Datenvolumen. Die Einschränkung ist von mir getroffen, weil sonst die mit dieser Problematik verbundene Komplexität den Rahmen dieser Arbeit sprengt.



**Abbildung 19: Struktur eines einfachen Materials**

Wenn ein Material transportiert werden soll, so versendet das Postversandsystem alle technisch referenzierten Sub-Materialien ebenfalls mit. Diese Vorgehensweise ist bei Materialien mit der in Abbildung 19 dargestellten Struktur unkritisch, da die Sub-Materialien integrale Bestandteile des Materials sind und für sich allein genommen keine Existenzberechtigung haben. Außerdem wäre das Material unvollständig, wenn es ohne seine Sub-Materialien versendet werden würde.

## Welche Probleme können beim Austausch der Materialien auftreten?

Beim Austausch von Materialien zwischen Arbeitsumgebungen durch das Postversandsystem zeigen sich folgende Probleme:

### 1. *Der Typ des Materials geht verloren*

Da der Aspekt `TransportierbaresMaterial` von der konkreten Implementation abstrahiert, kennt das Postversandsystem das Material nur unter diesem Typ. Auf der Seite des Absenders ist das unkritisch, da er weiß, was für ein Material er versendet. Auf der Seite des Empfängers sieht das anders aus. Er empfängt das Material unter dem Aspekt `TransportierbaresMaterial`. Was für ein konkretes Material sich hinter diesem Aspekt verbirgt, bleibt für ihn ungewiß. Der Aspekt, der sich für das Postversandsystem als vorteilhaft darstellte, erweist sich nun für den Empfänger als Nachteil. Das Problem läßt sich nur so beheben, indem die Typinformation des Materials mitgesendet wird. Diese kann dann auf der Empfängerseite erfragt werden, um zu ermitteln, um welchen Materialtyp es sich handelt. Da das Postversandsystem in der Programmiersprache JAVA implementiert ist, tritt das beschriebene Problem in diesem Ausmaß nicht auf. Bei dieser Programmiersprache kann jedes Objekt nach seinem Typ befragt werden. So kann der Empfänger ein Material befragen, das er nur unter dem Aspekt `TransportierbaresMaterial` kennt, ob es sich z.B. um eine Vorgangsmappe handelt. In anderen Programmiersprachen, die eine solche Funktionalität nicht bieten, muß diese nachträglich hinzugefügt werden.

## 2. *Der Transport komplexer Materialien*

Bei meinen bisherigen Ausführungen bin ich von einfach strukturierten Materialien ausgegangen. Es stellt sich jedoch die Frage, wie der Transport zu realisieren ist, wenn ein Material komplex ist. Dabei verstehe ich unter einem komplexen Material, wenn es mit vielen weiteren Materialien in Beziehung steht bzw. es ein sehr großes Datenvolumen hat. Ich möchte an dieser Stelle diese Problematik nicht ausführlich diskutieren, sondern lediglich grob umreißen und mögliche Lösungsansätze skizzieren.

Wenn es sich bei der Beziehung zwischen den Materialien um eine Kompositionsbeziehung handelt (siehe Abbildung 19), stellt das kein Problem dar. Haben jedoch die Materialien eine einfache Beziehung zueinander, da jedes für sich genommen eigenständig ist, so verbietet sich die Verknüpfung der Materialien mit Hilfe von technischen Referenzen. Diese technischen Referenzen hätten zur Folge, daß alle in Beziehung stehenden Materialien unerwünscht versendet werden würden.

Ein Lösungsansatz dafür wäre, die Materialien mittels fachlicher Referenzen, z.B. in Form von Identifikatoren, zu verknüpfen. Jedoch muß das bereits bei der Modellierung der Materialien berücksichtigt werden. Ein Beispiel für die Verwendung einer fachlichen Referenz wäre folgendes.

Der Kunde einer Bank kann über ein Konto verfügen. Das modellierte Material Kunde verweist jedoch nicht mittels einer technischen Referenz auf das zugehörige Material Konto, sondern besitzt als eine fachliche Referenz die Kontonummer. Diese hat identifizierenden Charakter und ist eindeutig. Wenn nun das Konto zu einem Kunden benötigt wird, so kann es bei einer dafür zuständigen Instanz unter Angabe der Kontonummer besorgt werden.

Bei diesem Lösungsansatz würden nicht die referenzierten Materialien übertragen werden, sondern nur die fachlichen Referenzen. Dabei ergibt sich das nächste Problem, daß das fachlich referenzierte Material auf der Empfängerseite über einen anderen Mechanismus bei Bedarf zugreifbar sein muß.

Das evtl. große Datenvolumen komplexer Materialien stellt ein weiteres Problem dar. Als Lösung ist vorstellbar, daß nicht das Material selbst versendet wird, sondern ein leichtgewichtiger Stellvertreter (vgl. [GHJ+95], S. 207ff.). Dieser Stellvertreter stellt die angeforderten Daten über einen Mechanismus zur Verfügung, ohne sie selbst zu enthalten.

## **5.3 Komponenten**

In diesem Abschnitt möchte ich die einzelnen Komponenten erörtern, die im Entwurf eingeführt wurden. Doch zuerst möchte ich konkretisieren, was ich unter einer Komponente verstehe. Dabei belasse ich es bei einem sehr einfachen Modell, obwohl mir bewußt ist, daß die Problematik der Komponenten weit komplexer ist. Jedoch genügt dieses einfache Modell von Komponenten völlig, um das Postversandsystem in seinem Aufbau zu verstehen.

### **Was verstehe ich unter einer Komponente?**

Unter einer Komponente verstehe ich eine in sich abgeschlossene Sinneinheit. Sie kapselt ihre Implementation und ist über definierte Schnittstellen in einem Kontext eingebunden. Mit

diesem Verständnis von einer Komponente orientiere ich mich an Szyperski. Er hält bezüglich Komponenten folgendes fest:

"[...] a component needs to encapsulate its implementation and interact with its environment through welldefined interfaces." ([Szy98], S. 30)

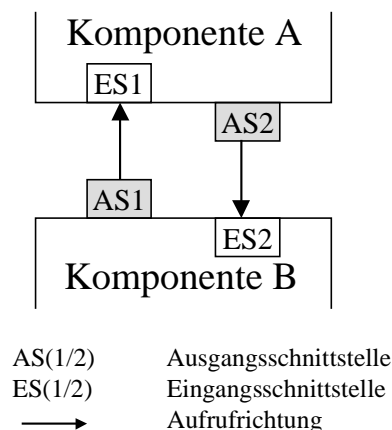
Dieses Verständnis von Komponenten wende ich zur *Strukturierung von Rahmenwerken* an. Somit verstehe ich im Rahmen dieser Arbeit unter einer Komponente einen abgeschlossenen Teil eines gesamten Rahmenwerkes, der eine definierte Aufgabe erfüllt und nur über spezielle explizite Schnittstellen in einem größeren Kontext eingebunden ist.

Dabei ist hervorzuheben, daß eine Komponente nicht nur eine Schnittstelle haben kann. Insbesondere unterscheide ich zwischen zwei Arten von Schnittstellen, den Ein- und den Ausgangsschnittstellen. Diese Schnittstellen sind aus der Sicht der Komponente motiviert.

Die Eingangsschnittstelle enthält die Umgangsformen, welche die Komponente nach außen hin anbietet. Über diese Schnittstelle können Operationen der Komponente aufgerufen werden. Aus der Sicht der Komponente gelangen über die Eingangsschnittstelle sozusagen die Aufrufe in sie hinein.

Damit eine Komponente auch die Dienste eine anderen Komponente in Anspruch nehmen kann, verfügt sie über Ausgangsschnittstellen. Über eine Ausgangsschnittstelle gehen sozusagen die Aufrufe aus der Sicht der Komponente nach außen.

Mit diesem Konzept der Ein- und Ausgangsschnittstellen werden die Beziehungen und Abhängigkeiten der Komponenten untereinander explizit gemacht. Weiterhin läßt sich feststellen, daß in einer komponentenorientierten Architektur die Ausgangsschnittstelle einer Komponente gleichzeitig die Eingangsschnittstelle einer anderen Komponente sein muß. In Abbildung 20 ist dargestellt, wie Komponenten über ihre Schnittstellen in Beziehung stehen.



**Abbildung 20: Darstellung von Komponenten mit ihren Ein- und Ausgangsschnittstellen**

Die Abbildung zeigt, daß z.B. die Schnittstelle 1 bei der Komponente B eine Ausgangsschnittstelle ist. Gleichzeitig wird sie als Eingangsschnittstelle von Komponente A angeboten. Dadurch ist es möglich, daß die Komponente B über ihre Ausgangsschnittstelle 1 Operationen der Komponente A aufrufen kann.

Zusammenfassend läßt sich festhalten, daß die Ein- und Ausgangsschnittstellen als Konnektoren fungieren, über die Komponenten kombiniert werden können.

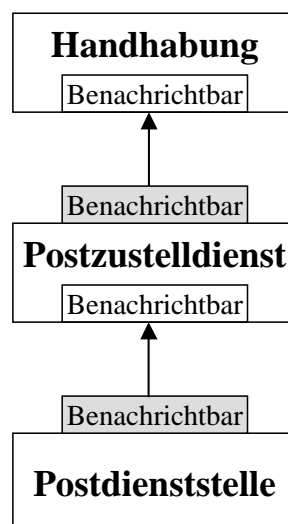
Neben den Ein- und Ausgangsschnittstellen der Komponenten gibt es noch weitere Schnittstellen, die sich nicht direkt einer Komponente zuordnen lassen. Sie dienen nicht als Schnittstelle für Aufrufe von Operationen an Komponenten, sondern als Parameter. Da diese Schnittstellen nicht in der Verantwortung einer konkreten Komponente sind, halte ich fest, daß sie allgemein im Anwendungssystem bekannt sein müssen. Das Postversandsystem hat eine Reihe solcher Schnittstellen, die bereits im Rahmen verschiedener Fragestellungen bei der Konzeption im Abschnitt 5.2 diskutiert wurden.

Bevor ich auf die einzelnen Komponenten eingehe, möchte ich noch eine Fragestellung diskutieren, die alle Komponenten des Postversandsystems betrifft.

### **Wie kann zwischen den Komponenten die Benachrichtigung übertragen werden?**

Der Entwurf (siehe Abschnitt 5.1) sieht vor, daß das Postversandsystem sich in drei Komponenten unterteilt. Das Problem ist, daß die Postdienststelle Benachrichtigungen an die Handhabungskomponente des Empfängers sendet und somit die Benachrichtigung in Form der Transportbeschreibung des erhaltenen Materials von Komponente zu Komponente übertragen werden muß.

Wie ich oben schon erwähnt habe, verfügen Komponenten über Ein- und Ausgangsschnittstellen. Um nun die Benachrichtigung zwischen den Komponenten auszutauschen, müssen dafür entsprechende Schnittstellen vorhanden sein. Zu diesem Zweck führe ich die Schnittstelle `Benachrichtbar` ein. Abbildung 21 zeigt, wie diese Schnittstelle in den verschiedenen Komponenten als Ein- und Ausgangsschnittstelle verwendet wird.



**Abbildung 21: Verwendung der Schnittstelle `Benachrichtbar`**

Die Postdienststelle informiert über die Ausgangsschnittstelle `Benachrichtbar` den Postzustelldienst. Dieser erhält die Nachricht in Form einer Transportbeschreibung über seine Eingangsschnittstelle und transportiert sie zum Empfänger. Dort gibt der Postzustelldienst

diese Nachricht über die Ausgangsschnittstelle der Handhabungskomponente. Die Handhabungskomponente erhält diese Benachrichtigung über ihre Eingangsschnittstelle und verarbeitet sie.

```
Benachrichtbar
-----
reagiere(Transportbeschreibung)
```

### Abbildung 22: Umgangsform der Schnittstelle Benachrichtbar

Die Schnittstelle `Benachrichtbar` hat lediglich eine Umgangsform (siehe Abbildung 22). Diese erlaubt, die Transportbeschreibung eines Materials zu übergeben. Wie die benachrichtigte Komponente darauf reagiert, ist verschieden. So transportiert der Postzustelldienst diese Transportbeschreibung von der Postdienststelle zum Empfänger, während die Handhabungskomponente in den meisten Fällen das entsprechende Material beim Postzustelldienst anfordert.

Im folgenden werde ich nun die einzelnen Komponenten erörtern. Dabei gehe ich zuerst immer auf die Ein- und Ausgangsschnittstellen ein, bevor ich beschreibe, wie die Komponente intern aufgebaut ist. Anhand des internen Aufbaus wird deutlich, wie das Postversandsystem realisiert ist und wie damit den verschiedenen Anforderungen entsprochen wird. An dieser Stelle möchte ich noch einmal betonen, daß der interne Aufbau aus der Sicht der Komponentenorientierung prinzipiell unerheblich ist, solange den Schnittstellen genügt wird.

#### 5.3.1 Handhabung

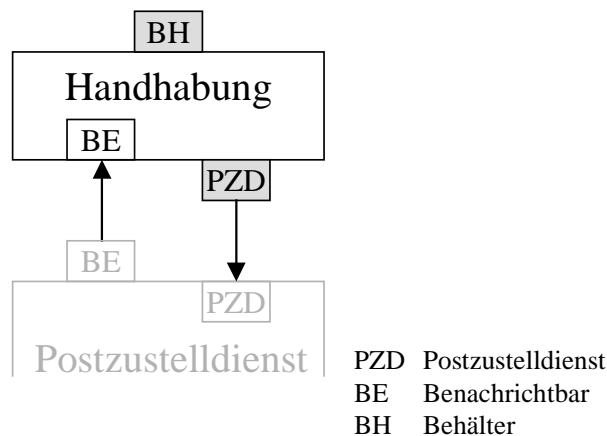
Die Handhabungskomponente realisiert das Benutzungsmodell des Postversandsystems. Da es eine Anforderung ist, beliebige Benutzungsmodelle zu unterstützen, stellt die Handhabungskomponente einen Hot-Spot dar. Je nach Anwendungskontext können verschiedene Handhabungskomponenten eingesetzt werden, die den jeweiligen Anforderungen angemessen sind. Wichtig hierbei ist lediglich, daß sie über die Eingangsschnittstelle `Benachrichtbar` (siehe oben) und die Ausgangsschnittstelle `Postzustelldienst` (siehe Abschnitt 5.3.2) verfügt, um in das Postversandsystem eingebunden werden zu können.

Da die Handhabungskomponente den Umgang des Benutzers mit dem Postversandsystem bestimmt, ist sie lokal auf dem Arbeitsplatz des Benutzers angesiedelt.

In diesem Abschnitt stelle ich eine Handhabungskomponente vor, welche die Benutzungsmodelle aus Abschnitt 4.3 realisiert. Daher beschränkt sich auch die Funktionalität auf das für die Benutzungsmodelle Notwendige. Sie hat, wie in Abbildung 23 dargestellt, als Eingangsschnittstelle `Benachrichtbar` und als Ausgangsschnittstellen `Postzustelldienst` und `Behälter`<sup>15</sup>.

---

<sup>15</sup> Die Schnittstelle `Behälter` ist kein Bestandteil des Postversandsystems, sondern ist durch das JWAM-Rahmenwerk gegeben. Sie hat Umgangsformen für das Ablegen, Entnehmen und Sondieren von Materialien.



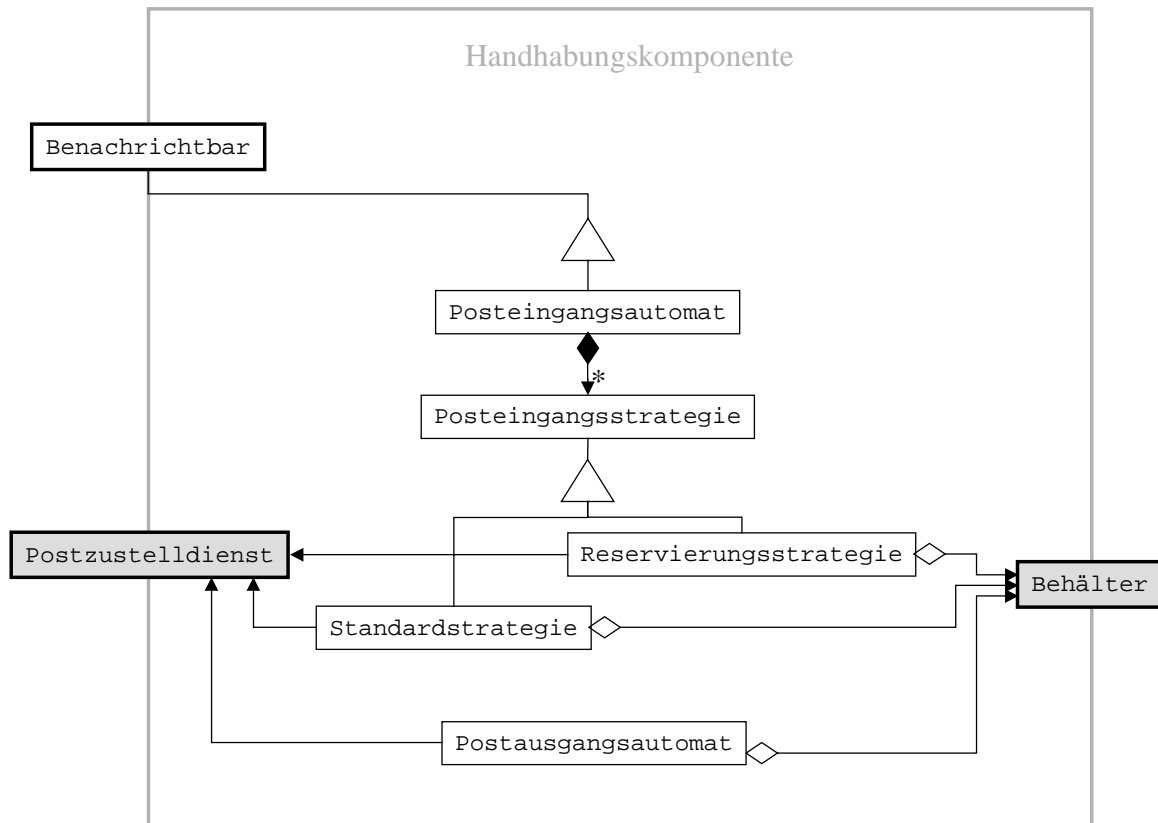
**Abbildung 23: Ein- und Ausgangsschnittstellen der Handhabungskomponente**

Über die Eingangsschnittstelle Benachrichtbar wird die Handhabungskomponente vom Postzustelldienst über den Erhalt eines Materials informiert. Um diese Benachrichtigung zu bearbeiten, kann die Handhabungskomponente über die Ausgangsschnittstelle Postzustelldienst das Material anfordern. Weiterhin können über diese Ausgangsschnittstelle auch Materialien versendet werden. Die Ausgangsschnittstelle Behälter wird von der Handhabungskomponente dazu verwendet, auf die Behälter zuzugreifen, die als Postkörbe fungieren. So werden entweder Materialien in die Postkörbe abgelegt, wenn es sich um Posteingangskörbe handelt, bzw. entnommen, wenn es Postausgangskörbe sind.

Ich möchte an dieser Stelle nochmals darauf hinweisen, daß beliebige Benutzungsmodelle denkbar wären. Es müßte dann lediglich eine für den jeweiligen Anwendungsfall passende Handhabungskomponente geschaffen werden, die als Eingangsschnittstelle Benachrichtbar und als Ausgangsschnittstelle Postzustelldienst hat.

Die Abbildung 24 zeigt einen Überblick, wie die Handhabungskomponente realisiert ist. Die Ein- und Ausgangsschnittstellen sind schwarz hervorgehoben.

In den folgenden Abschnitten werde ich die einzelnen Bestandteile der Handhabungskomponente im Detail erläutern und dabei die entsprechenden Teile der Architektur erneut aufgreifen.



**Abbildung 24: Aufbau der Handhabungskomponente**

### 5.3.1.1 Posteingangsautomat

Im Entwurf (siehe Abschnitt 5.1) habe ich für den Empfang der Materialien auf dem Arbeitsplatz des Benutzers einen Posteingangsautomaten vorgesehen. Er regelt, wie auf die Benachrichtigung über den Neuzugang eines Materials im Postfach der Postdienststelle zu reagieren ist. Er ist damit die entscheidende Stelle, die den Materialempfang als Teil des Benutzungsmodells (siehe Abschnitt 4.3) bestimmt.

Eine Anforderung an den Posteingangsautomaten ist, daß er flexibel an verschiedene Bedürfnisse des Benutzers angepaßt werden kann. Um dieser Anforderung gerecht zu werden, läßt er sich mit Posteingangsstrategien (siehe Abschnitt 5.3.1.2) konfigurieren. Diese Entwurfslösung orientiert sich an dem Entwurfsmuster "Strategy" (vgl. [GHJ+95], S. 315ff.).

Es wäre auch ein wesentlich einfacherer Aufbau denkbar, bei dem der Posteingangsautomat direkt auf die Benachrichtigungen reagiert, ohne dabei auf Strategien zurückzugreifen. Allerdings wären unter diesen Umständen die verschiedenen in Abschnitt 4.3 geforderten Benutzungsmodelle nur realisierbar, wenn je ein spezialisierter Posteingangsautomat geschaffen würde. Das allerdings widerspricht der Anforderung nach Basisfunktionalität aus Abschnitt 4.4.

Der erweiterte Umgang mit dem Postversandsystem (siehe Abschnitt 4.3.2) sieht vor, daß sich auf dem Arbeitsplatz des Benutzers mehrere Posteingangskörbe befinden können. Dabei ergibt sich das Problem, daß es überschneidende Zuständigkeiten geben kann. Ich meine

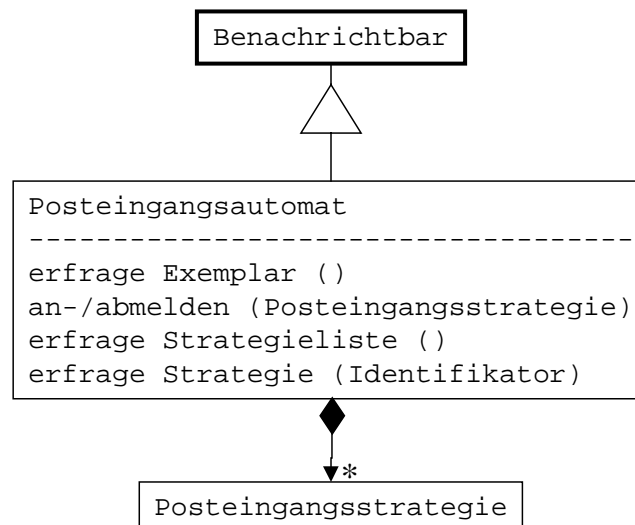


damit, daß ein empfangenes Material in verschiedene Posteingangskörbe abgelegt werden könnte.

Um dieses Problem zu lösen, sehe ich vor, daß die Posteingangskörbe in einer definierbaren Reihenfolge bedient werden. Die von mir vorgeschlagene Entwurfslösung des Posteingangsautomaten, löst dieses Reihenfolgeproblem. Dazu sehe ich vor, daß es in der Arbeitsumgebung des Benutzers nur einen Posteingangsautomaten geben darf. Daraus folgt, daß der Posteingangsautomat als ein Singleton (vgl. [GHJ+95], S. 127ff.) realisiert ist. Der Posteingangsautomat bestimmt die Reihenfolge, in der die Strategien aktiviert werden, die den jeweiligen Posteingangskorb bedienen.

Es stellt sich nun die Frage, wie die Reihenfolge definiert wird. Meine Lösung des Problems sieht vor, daß die Strategien in der Reihenfolge bedient werden, in welcher der Posteingangsautomat mit den Strategien konfiguriert wurde. Diese Reihenfolge wird auch in der erfragbaren Strategieliste widerspiegelt. Es liegt somit in der Verantwortung des Administrators, eine den Bedürfnissen angepaßte Reihenfolge bei der Konfiguration mit dem Einstellwerkzeug (siehe Abschnitt 5.4.4) zu berücksichtigen.

In Abbildung 25 ist der Aufbau des Posteingangsautomaten dargestellt. Er erfüllt die Eingangsschnittstelle Benachrichtbar (siehe Abschnitt 5.3) der Handhabungskomponente und hat eine beliebige Anzahl von Posteingangsstrategien.



**Abbildung 25: Aufbau des Posteingangsautomaten**

Die Umgangsformen des Posteingangsautomaten dienen der Konfiguration. So bietet er die Möglichkeit, konkrete Posteingangsstrategien an- bzw. abzumelden. Eine Übersicht über alle registrierten Posteingangsstrategien sowie deren Identifikatoren ist erhältlich, indem die Strategieliste erfragt wird. Weiterhin besteht auch die Möglichkeit, eine angemeldete Posteingangsstrategie unter Angabe des Identifikators zu erfragen. Diese Umgangsform wird benötigt, um mit einem entsprechenden Einstellwerkzeug die Einstellungen einer konkreten Strategie zu modifizieren.

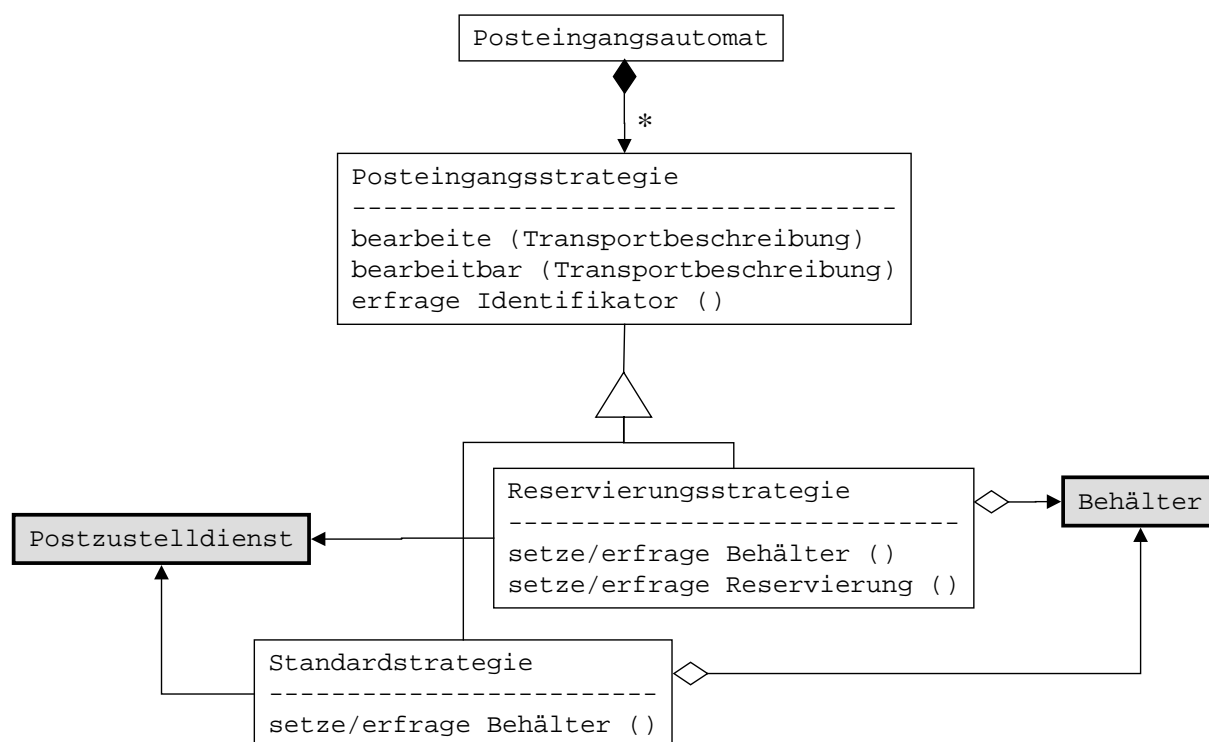
Wie bereits erwähnt, ist der Posteingangsautomat als ein Singleton realisiert. Somit bietet die gleichnamige Schnittstelle die Möglichkeit, das einzige Exemplar des Posteingangsautomaten zu erfragen.

### 5.3.1.2 Posteingangsstrategien

Die Posteingangsstrategie verkörpert den Algorithmus, ob und wie auf eine Benachrichtigung des Postzustelldienstes (siehe Abschnitt 5.3.2) in Form einer Transportbeschreibung zu reagieren ist. Sie wird für die Konfiguration des Posteingangsautomaten (siehe oben) verwendet. Neben dem Posteingangsautomaten greifen auch entsprechende Einstellwerkzeuge (siehe Abschnitt 5.4.4) auf die Posteingangsstrategien zu.

Um an den jeweiligen Anwendungskontext angepaßte Posteingangsstrategien zu ermöglichen, konzipiere ich die Klasse `Posteingangsstrategie` als einen Hot-Spot (siehe Abschnitt 4.4.3). Sie repräsentiert die minimale Schnittstelle, die eine beliebige Realisierung aufweisen muß, damit sie vom Posteingangsautomaten verwendet werden kann. Dadurch ist es möglich, zu den in diesem Abschnitt vorgestellten Strategien weitere hinzuzufügen.

Der Aufbau sowie die Umgangsformen sind in Abbildung 26 dargestellt.



**Abbildung 26: Aufbau der Posteingangsstrategien**

Der Posteingangsautomat befragt die Posteingangsstrategie, ob sie eine Benachrichtigung in Form einer Transportbeschreibung bearbeiten kann. Entscheidet sich die Posteingangsstrategie positiv, so beauftragt sie der Posteingangsautomat mit der Bearbeitung. Weiterhin bietet die Posteingangsstrategie einen Identifikator an. Dieser wird zur Verwaltung aller Strategien beim Posteingangsautomaten benötigt.

In Anlehnung an die in Abschnitt 4.3 geforderten Benutzungsmodelle habe ich bei den spezialisierten Posteingangsstrategien, wie die Standard- bzw. die Reservierungsstrategie, die Bearbeitung einer Benachrichtigung in Form einer Transportbeschreibung wie folgt vorgesehen. Sie greifen auf den Postzustelldienst über die gleichnamige Ausgangsschnittstelle zu, um sich anhand der Transportbeschreibung das empfangene Material aus dem Postfach zu

besorgen. Anschließend legen sie das Material in den ihr zugeordneten Postkorb über die Ausgangsschnittstelle Behälter ab. Die Standard- bzw. die Reservierungsstrategie unterscheiden sich lediglich in der Auswahl, welche Benachrichtigungen in Form von Transportbeschreibungen sie bearbeiten.

Ich beschränke mich beispielhaft auf die folgenden beiden Posteingangsstrategien, da andere Varianten über die Anforderungen des Benutzungsmodells, wie es in Abschnitt 4.3 beschrieben ist, hinausgehen würden.

### **Standardstrategie**

Die Standardstrategie bearbeitet alle Benachrichtigungen. Aus diesem Grund sollte sie, falls der Posteingangsautomat mit mehreren Strategien konfiguriert wird, in der Auswahlreihenfolge die letzte sein.

### **Reservierungsstrategie**

Die Reservierungsstrategie bearbeitet nur solche Benachrichtigungen, bei denen der in der Transportbeschreibung erfragbare Absender mit der Reservierung identisch ist. Diese Strategie sollte sich in der Auswahlreihenfolge des Posteingangsautomaten immer vor der Standardstrategie befinden.

## **5.3.1.3 Postausgangsautomat**

Im Entwurf (siehe Abschnitt 5.1) ist für die Versendung von Materialien aus Postausgangskörben der Postausgangsautomat vorgesehen. Er bestimmt, wie versendet werden soll. Der in diesem Abschnitt beschriebene Postausgangsautomat entspricht den Benutzungsmodellen aus Abschnitt 4.3.

Da es beim Postausgangsautomaten nicht das Reihenfolgeproblem bei der Bedienung der einzelnen Postkörbe gibt, wie beim Posteingangsautomaten (siehe Abschnitt 5.3.1.1), ist der Aufbau wesentlich einfacher. So sehe ich pro Behälter, der als Postausgangskorb fungiert, genau einen Postausgangsautomaten vor. Somit kann es beliebig viele Postausgangsautomaten in der Arbeitsumgebung des Benutzers geben.

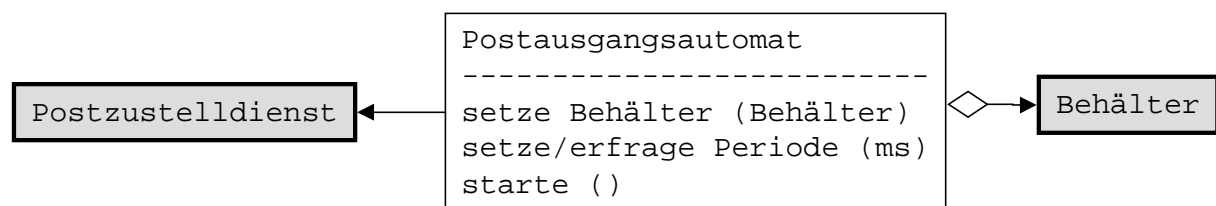
Der Postausgangsautomat wird von keiner Komponente des Postversandsystems verwendet. Aus diesem Grund ist es auch nicht notwendig, ihn als Hot-Spot zu konzipieren. Das bedeutet nicht, daß es nicht beliebige anwendungsspezifische Postausgangsautomaten geben kann. Vielmehr erübrigt sich lediglich, eine Schnittstelle vorzugeben, die ein spezieller Postausgangsautomat zu implementieren hat.

Der Postausgangsautomat arbeitet autonom in einstellbar periodischen Abständen und kann auch manuell gestartet werden. Dazu entnimmt er ein Material aus dem ihm zugeordneten Behälter und setzt die Absenderadresse, sofern sie noch nicht vorhanden ist. Anschließend befragt er den Postzustelldienst, ob es versendet werden kann. Ist das der Fall, so übergibt er es dem Postzustelldienst. Ist das Material jedoch nicht versendbar, so legt er es in den Postausgangskorb wieder zurück. Danach nimmt er das nächste Material und wiederholt den Versendevorgang solange, bis er alle Materialien versucht hat, zu versenden.

Prinzipiell kann es beliebig verschiedene Varianten des Postausgangsautomaten geben, die sich in ihrer Funktionsweise unterscheiden. So wäre auch ein Postausgangsautomat denkbar, der, bevor er ein Material versendet, eine Kopie in einen anderen Behälter ablegt. In dieser

Arbeit beschränke ich mich jedoch auf die oben beschriebene Variante, da mit deren Funktionalität bzw. Arbeitsweise den geforderten Benutzungsmodellen (siehe Abschnitte 4.3.1 und 4.3.2) hinsichtlich des Versendens von Materialien entsprochen wird.

Der Postausgangsautomat bedient die beiden Ausgangsschnittstellen Postzustelldienst und Behälter der Handhabungskomponente. Über die Schnittstelle Behälter greift er auf die Postausgangskörbe zu, um sich die zu versendenden Materialien zu besorgen. Anschließend versendet er die Materialien über die Ausgangsschnittstelle Postzustelldienst. In Abbildung 27 sind diese Abhängigkeiten sowie die Umgangsformen ersichtlich.



**Abbildung 27: Umgangsformen des Postausgangsautomaten**

Dem Postausgangsautomat kann ein Behälter zugeordnet werden, den er zu bedienen hat. Weiterhin kann an ihm die Periode in Millisekunden gesetzt und erfragt werden, in der er aktiv wird und alle Materialien aus dem ihm zugeordneten Behälter versendet. Damit der Automat auch manuell gestartet werden kann, wie es das erweiterte Benutzungsmodell (siehe Abschnitt 4.3.2) vorsieht, bietet er eine entsprechende Methode an.

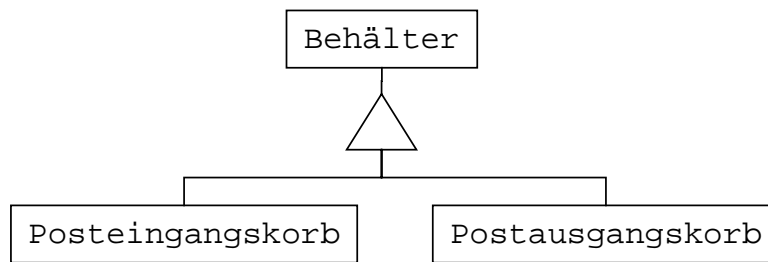
### 5.3.1.4 Postein- und Postausgangskorb

Im Benutzungsmodell aus Abschnitt 4.3.2 wird gefordert, daß die Postkörbe visuell signalisieren, ob sie leer oder befüllt sind. Aus diesem Grund habe ich die in diesem Abschnitt beschriebenen Postkörbe als optionale Bestandteile der Handhabungskomponente eingeführt. Sie erfüllen die Schnittstelle Behälter und zeigen über ihre Icons an, ob sie Materialien enthalten. Ich differenziere zwischen dem Postein- und dem Postausgangskorb, die sich nur in ihrer Repräsentation mittels Icon (siehe Abbildung 28) unterscheiden.



**Abbildung 28: Icons der Postkörbe (von links nach rechts: leerer und befüllter Postausgangskorb, leerer und befüllter Posteingangskorb)**

Neben der Schnittstelle Behälter haben die Postkörbe keine zusätzlichen Umgangsformen (siehe Abbildung 29). Allerdings hat der Postausgangskorb eine zusätzliche Eigenschaft. Es können in ihm nur Materialien abgelegt werden, die auch die Schnittstelle TransportierbaresMaterial erfüllen. Damit wird erreicht, daß sich in einem Postausgangskorb nur Materialien befinden können, die über das Postversandssystem auch versendbar sind.



**Abbildung 29: Aufbau des Postein- und Postausgangskorbes**

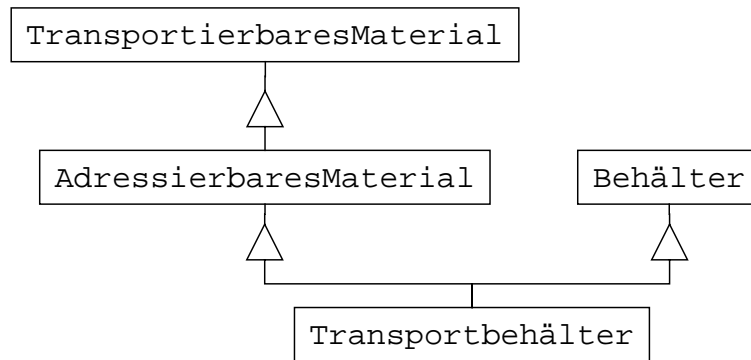
### 5.3.1.5 Transportbehälter

In der Anwendungswelt hat man es oftmals mit Materialien zu tun, die zwar ortsveränderbar aber nicht transportierbar im Sinne des Postversandsystems (vgl. Abschnitt 5.2.2) sind. Trotzdem sollen auch solche Materialien mit dem Postversandsystem versendet werden können. Zu diesem Zweck bietet sich die Verwendung von Behältern an, die ortsveränderbare Materialien aufnehmen und mit dem Postversandsystem transportiert werden können. Folgendes Beispiel hierfür.

Auf dem elektronischen Arbeitsplatz eines Bankangestellten befindet sich ein Kreditantrag. Dieses Material ist anwendungsfachlich modelliert worden, jedoch nicht als transportierbares Material im Sinne des Postversandsystems. Es können somit auf diesem Antrag Angaben zum Antragsteller sowie die Kredithöhe etc. vermerkt werden, jedoch keine Transportinformationen. Der Kreditantrag hat zwar die Eigenschaft, daß er ortsveränderbar ist, aber er wurde nicht für den Transport mit dem Postversandsystem explizit vorgesehen. Wenn nun der Bankangestellte den Kreditantrag an einen anderen Mitarbeiter weiterleiten möchte, so könnte er ihn nicht unmittelbar mit dem Postversandsystem senden. Vielmehr hat er jedoch die Möglichkeit, diesen Antrag in einen Transportbehälter abzulegen. Der Transportbehälter ist transportierbar im Sinne des Postversandsystems und kann ortsveränderbare Materialien aufnehmen. Diesen Transportbehälter kann nun der Bankangestellte mit dem Postversandsystem zu dem gewünschten Empfänger senden.

Der Transportbehälter ist ein solcher Behälter. Er kann ortsveränderbare Materialien aufnehmen und erfüllt die Schnittstelle AdressierbaresMaterial (siehe Abschnitt 5.2.2). Aus diesem Grund ist es auch möglich, ihn mit dem Postversandsystem zu versenden. In Abbildung 30 ist der Aufbau des Transportbehälters dargestellt.

Da der Transportbehälter unverändert nutzbar ist und er die Möglichkeit schafft, bereits im Anwendungssystem vorhandene ortsveränderbare Materialien ohne weitere Anpassungen versenden zu können, trägt er zur Erfüllung der Anforderung aus Abschnitt 4 bei, das Postversandsystem als Groupware einsetzen zu können.

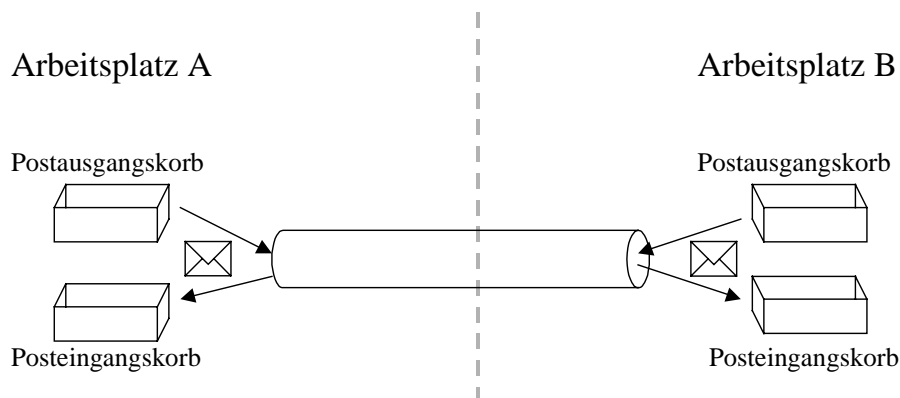


**Abbildung 30: Aufbau des Transportbehälters**

### 5.3.1.6 Exkurs: Rohrpost

In diesem Abschnitt beschreibe ich, wie durch einfache Anpassung der Handhabungskomponente eine Rohrpost realisierbar ist. Dadurch wird deutlich, daß das Postversandssystem nicht nur den in Abschnitt 4.3 geforderten Benutzungsmodellen entspricht, sondern darüber hinaus an weitere Anwendungskontexte angepaßt werden kann und somit der Forderung nach Basisfunktionalität aus Abschnitt 4.4 genügt.

Eine Rohrpost zeichnet aus, daß sie eine direkte Verbindung zwischen genau zwei Arbeitsplätzen darstellt. Dadurch können Materialien dem Empfänger zugesendet werden, ohne sie vorher explizit zu adressieren. Den Benutzern ist immer bewußt, von wem ein Material empfangen wurde bzw. an wen es versendet wird. In Abbildung 31 werden der Aufbau und die Funktionsweise verdeutlicht.



**Abbildung 31: Aufbau und Funktionsweise der Rohrpost**

Auf dem Arbeitsplatz befinden sich je ein Postein- und ein Postausgangskorb. Alle Materialien, die der Benutzer in den Postausgangskorb ablegt, werden auf Wunsch sofort oder aber in periodischen Abständen dem Empfänger übermittelt. Beim Empfänger werden sie dann im Posteingangskorb abgelegt.

Wichtig hierbei ist, daß aus der Sicht des Benutzers die Materialien lediglich ortsveränderbar und nicht mehr transportierbar im Sinne des Postversandsystems (siehe Abschnitt 5.2.2) sein müssen. Der Grund dafür ist, daß durch die Rohrpost der Empfänger sowie der Absender festgelegt sind und somit die Transportinformationen am Material selbst überflüssig sind.

Um eine Rohrpost zu realisieren, sind Anpassungen an den folgenden zwei Stellen der Handhabungskomponente nötig:

- *Postausgangsautomat*  
Für die Zustellung werden der Postzustelldienst (siehe Abschnitt 5.3.2) und die Postdienststelle (siehe Abschnitt 5.3.3) verwendet. Allerdings können diese Komponenten nur Materialien übertragen, die auch die Schnittstelle TransportierbaresMaterial (siehe Abschnitt 5.2.2) erfüllen. Somit ist es erforderlich, daß die in dem Postausgangskorb befindlichen Materialien in Transportbehälter (siehe Abschnitt 5.3.1.5) abgelegt und mit der Empfängeradresse versehen werden. Diese Aufgabe kann von einem spezialisierten Postausgangsautomaten vorgenommen werden, wenn er die Materialien aus dem ihm zugeordneten Postausgangskorb versendet.
- *Posteingangsstrategie*  
Für das Empfangen von Materialien sind im wesentlichen auf Empfängerseite die Posteingangsstrategien (siehe Abschnitt 5.3.1.2) des Posteingangsautomaten (siehe Abschnitt 5.3.1.1) zuständig. Es wird nun eine spezialisierte Posteingangsstrategie benötigt, welche die Materialien aus den empfangenen Transportbehältern entnimmt und in den entsprechenden Posteingangskorb ablegt. Da pro Arbeitsplatz mehrere Rohrpostverbindungen bestehen können, bietet sich die Reservierungsstrategie (siehe Abschnitt 5.3.1.2) zur Spezialisierung an, da sie in Abhängigkeit der Absenderadresse aktiv wird.

Mit den beschriebenen Anpassungen kann das in dieser Arbeit beschriebene Postversandsystem auf einfache Weise in eine Rohrpost umgebaut werden. Allerdings können Probleme auftreten, wenn zwischen zwei Arbeitsplätzen sowohl das Postversandsystem als auch eine Rohrpost zum Austausch von Materialien verwendet werden. Dann nämlich kann auf der Empfängerseite nicht mehr unterschieden werden, ob ein Material mit der Rohrpost verschickt wurde, oder nicht, da der Transport in beiden Fällen vom Postzustelldienst zusammen mit der Postdienststelle vorgenommen wird. Bei solchen zusätzlichen Anforderungen, müssen weitere Anpassungen vorgenommen werden. So wäre z.B. eine Lösung, spezielle Transportbehälter zu verwenden, anhand derer man auf Empfängerseite erkennen kann, ob das Material mit der Rohrpost gesendet wurde.

### 5.3.2 Postzustelldienst

Der Entwurf aus Abschnitt 5.1 sieht vor, daß es eine Postdienststelle (siehe Abschnitt 5.3.3) gibt, auf die zentral zugegriffen werden kann, während sich die Handhabungskomponenten (siehe Abschnitt 5.3.1) auf den Arbeitsplatzcomputern der jeweiligen Benutzer befinden. Aufgrund dieser Entwurfsentscheidung ergibt sich das Problem, daß die Komponenten auf verschiedenen Computern angesiedelt sein können, aber miteinander in Verbindung stehen müssen.

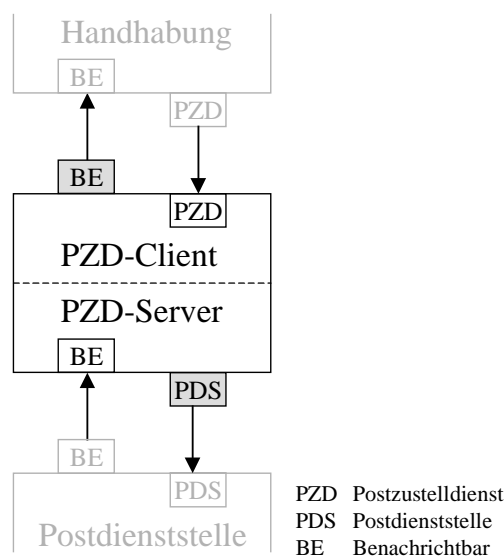
Der Postzustelldienst löst dieses Problem, indem er als Bindeglied zwischen den Komponenten Postdienststelle und Handhabung fungiert. Er stellt einen Hot-Spot dar, indem er von der konkret verwendeten Middleware abstrahiert, die für die Interprozeßkommunikation benötigt wird. Weiterhin hat er die Aufgaben, die Dienste der Postdienststelle der Handhabungskomponente zur Verfügung zu stellen sowie die

Benachrichtigungen der Postdienststelle der entsprechenden Handhabungskomponente zu übermitteln.

Ein weit verbreiteter Ansatz in der objektorientierten Softwareentwicklung, der sich für die Realisierung verteilter Anwendungen durchgesetzt hat, ist das Dienstleistungsprinzip (vgl. [GKZ94]). Es unterteilt die Komponenten eines Softwaresystems in Leistungsanbieter und -abnehmer. Der Leistungsanbieter stellt eine Dienstleistung den Leistungsabnehmern über eine Schnittstelle zur Verfügung. Dabei ist es unerheblich und bleibt dem Leistungsabnehmer verborgen, wie und wo die Dienstleistung erbracht wird.

Für die Konzeption des Postzustelldienstes bietet sich das Dienstleistungsprinzip zur Lösung des Verteilungsproblems an. Somit unterteilt sich die Komponente in den Postzustelldienst-Server (siehe Abschnitt 5.3.2.2) und -Client (siehe Abschnitt 5.3.2.1). Der Postzustelldienst-Server ist der Leistungsanbieter gegenüber dem Postzustelldienst-Client, während er gleichzeitig Leistungsabnehmer gegenüber der Postdienststelle ist. Der Postzustelldienst-Client ist Leistungsabnehmer des Postzustelldienst-Servers und gleichzeitig Leistungsanbieter für die Handhabungskomponente. Der Aufbau sowie die wechselseitigen Beziehungen werden in Abbildung 32 deutlich.

Durch die gewählte Unterteilung in Postzustelldienst-Server und -Client wird erreicht, daß die Handhabungskomponente die Dienstleistung des Postzustelldienst-Client in Anspruch nehmen kann, unabhängig davon, wie diese erbracht wird. Die Dienstleistung besteht in diesem Fall darin, auf die Postdienststelle über ein Netzwerk zugreifen zu können, unabhängig davon, auf welchem Computer sie sich befindet und welche Middleware für die Netzwerkkommunikation eingesetzt wird. Der Postzustelldienst-Client wiederum bedient sich der Dienstleistung des Postzustelldienst-Servers unter Verwendung einer konkreten Middleware. Der Postzustelldienst-Server erbringt seine Dienstleistung, indem er auf die Postdienststelle zugreift. Da der Postzustelldienst-Server Leistungsabnehmer der Postdienststelle ist, bleibt ihr verborgen, ob auf sie über ein Netzwerk verteilt zugegriffen wird oder nicht.



**Abbildung 32: Ein- und Ausgangsschnittstellen des Postzustelldienstes**



Um mit der Handhabungskomponente sowie der Postdienststelle kommunizieren zu können, verfügt der Postzustelldienst über die Eingangsschnittstellen Benachrichtbar und Postzustelldienst und über die Ausgangsschnittstellen Benachrichtbar und Postdienststelle (siehe Abbildung 32).

Über die Eingangsschnittstelle Benachrichtbar erhält der Postzustelldienst-Server von der Postdienststelle Nachrichten in Form von Transportbeschreibungen, die er an den jeweiligen Postzustelldienst-Client weiterleitet. Dieser wiederum gibt diese Nachrichten über die entsprechende Ausgangsschnittstelle an die Handhabungskomponente weiter. Über die Eingangsschnittstelle Postzustelldienst kann die Handhabungskomponenten beim Postzustelldienst-Client Anfragen stellen bzw. Materialien aus einem Postfach bei der Postdienststelle anfordern. Diese Anfragen bzw. Anforderungen werden zum Postzustelldienst-Server weitergeleitet, der sie dann an die Postdienststelle richtet. Hierbei besteht die Aufgabe des Postzustelldienstes nicht nur darin, die Anfragen und Anforderungen an die Postdienststelle zu übermitteln, sondern auch für die Übertragung von Objekten wie Materialien und Transportbeschreibungen zu sorgen.

```
Postzustelldienst
-----
sende Material (TransportierbaresMaterial)
hole Material (Transportbeschreibung)
ist sendbar (TransportierbaresMaterial)
Inhaltsverzeichnis (Transportadresse)
Adreßliste ()

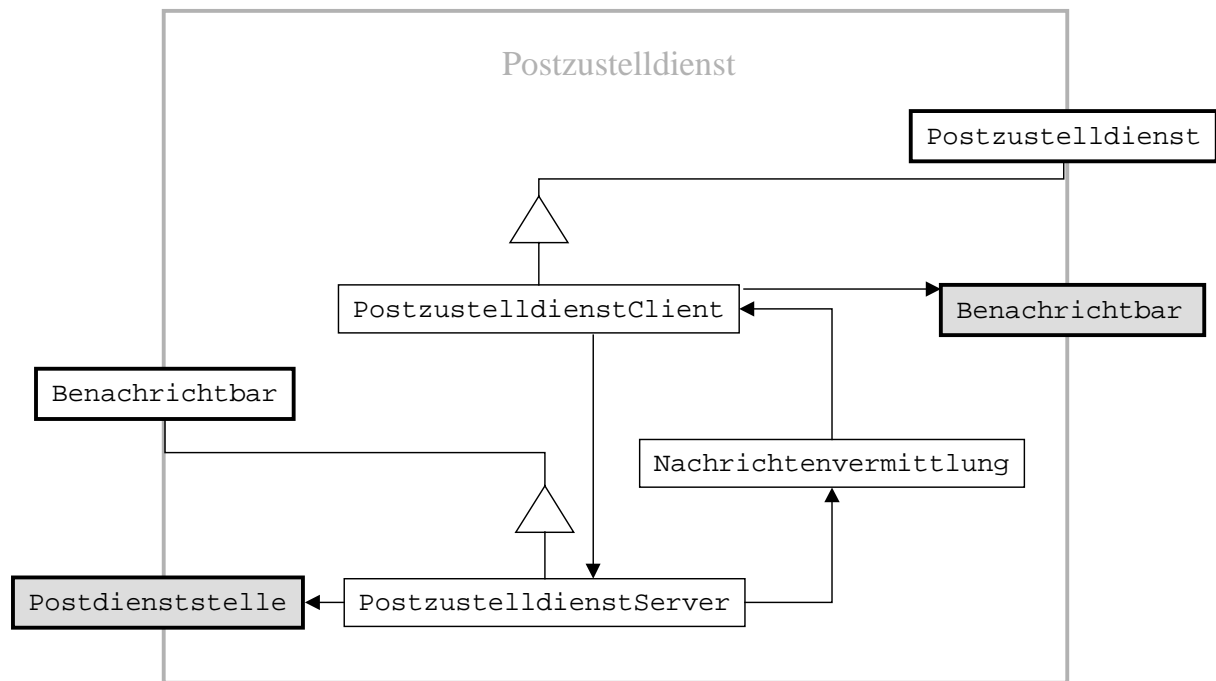
setze Arbeitsplatzadressen (Adreßliste)
an-/abmelden (Benachrichtbar)
```

### Abbildung 33: Umgangsformen des Postzustelldienstes

Da der Postzustelldienst im wesentlichen den verteilten Zugriff auf die Postdienststelle ermöglicht, sind die meisten seiner Umgangsformen mit ihr identisch. Aus diesem Grund erläutere ich die identischen Umgangsformen an dieser Stelle nicht und verweise auf den Abschnitt 5.3.3.

Darüber hinaus bietet der Postzustelldienst die Möglichkeit, daß sich eine Komponente mit der Eingangsschnittstelle Benachrichtbar an- bzw. abmeldet. Im Entwurf (siehe Abschnitt 5.1) ist die Handhabungskomponente dafür vorgesehen. Diese wird, wenn sie angemeldet ist, mit den Transportbeschreibungen benachrichtigt, die von der Postdienststelle versendet wurden.

Um zu bestimmen, für welche Postfächer die Handhabungskomponente benachrichtigt werden soll, kann eine Adreßliste vorgegeben werden. Sobald ein Material in ein entsprechendes Postfach bei der Postdienststelle abgelegt wird, erhält sie dann die Transportbeschreibung als Benachrichtigung.



**Abbildung 34: Aufbau des Postzustelldienstes**

Die Abbildung 34 zeigt einen Überblick, wie der Postzustelldienst realisiert ist. Die Ein- und Ausgangsschnittstellen sind schwarz hervorgehoben.

In den folgenden Abschnitten werde ich die einzelnen Bestandteile des Postzustelldienstes im Detail erläutern und dabei die entsprechenden Teile der Architektur erneut aufgreifen.

### 5.3.2.1 Postzustelldienst-Client

Der Postzustelldienst-Client realisiert die Eingangsschnittstelle `Postzustelldienst`. Er kapselt zusammen mit dem Postzustelldienst-Server die Interprozesskommunikation. Dabei nimmt er folgende Aufgaben wahr. Er leitet alle Aufrufe der Handhabungskomponente über die Eingangsschnittstelle `Postzustelldienst` an den Postzustelldienst-Server weiter. Da der Postzustelldienst-Server sich auf einem anderen Computer befindet, verwendet der Postzustelldienst-Client eine entsprechende Middleware. Ich habe bei der Realisierung des Postzustelldienstes als Middleware RMI verwendet.

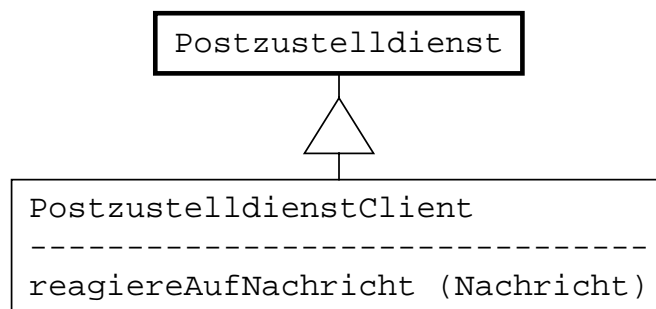
Die zweite Aufgabe des Postzustelldienst-Clients besteht darin, die Handhabungskomponente über die Ausgangsschnittstelle `Benachrichtbar` mittels der erhaltenen Transportbeschreibungen zu benachrichtigen. Diese Transportbeschreibungen erhält der Postzustelldienst-Client über eine Nachrichtenvermittlung<sup>16</sup> vom Postzustelldienst-Server.

Die Nutzung einer Nachrichtenvermittlung für den Informationsfluß vom Postzustelldienst-Server zum -Client ist technisch motiviert und hat den Vorteil, daß der Postzustelldienst-Server keinen Zugriff auf die Clients haben muß, was dem Dienstleistungsprinzip widersprechen würde. Weiterhin hat eine Nachrichtenvermittlung die Eigenschaft, die

<sup>16</sup> Eine Nachrichtenvermittlung überträgt Nachrichten asynchron zu beliebig vielen Interessenten. Die Nachrichten können oftmals beliebigen Inhalt haben. Im Falle des Postversandsystems enthalten sie die Transportbeschreibungen.

Transportbeschreibungen asynchron zu übertragen. Das ist insbesondere in diesem Fall wichtig, da sonst der Kontrollfluß beim Postzustelldienst-Server solange unterbrochen werden würde, bis die Transportbeschreibung beim Postzustelldienst-Client angekommen ist und bearbeitet wurde.

Eine Alternative zur Verwendung einer Nachrichtenvermittlung wäre, wenn der Postzustelldienst-Client periodisch den Postzustelldienst-Server abfragen würde, ob Transportbeschreibungen für ihn vorhanden sind. Dieses sogenannte Polling hat aber den Nachteil, daß es zu einem unnötigen Datenverkehr im Netzwerk kommt, was insbesondere bei einer großen Anzahl von Computerarbeitsplätzen zu Problemen führen kann.



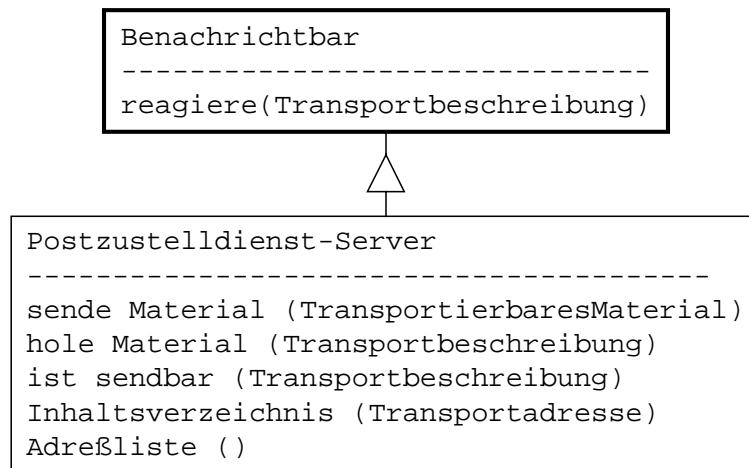
**Abbildung 35: Aufbau und Umgangsformen des Postzustelldienst-Client**

In Abbildung 35 sind die Umgangsformen des Postzustelldienst-Clients dargestellt. Er erweitert lediglich die Eingangsschnittstelle `Postzustelldienst` um eine Methode, die für die interne Benachrichtigung mit Hilfe einer Nachrichtenvermittlung benötigt wird.

### 5.3.2.2 Postzustelldienst-Server

Zur Kapselung der Interprozeßkommunikation zusammen mit dem Postzustelldienst-Client nimmt der Postzustelldienst-Server zwei Aufgaben wahr. Er delegiert die Aufrufe, die er vom Postzustelldienst-Client über eine Middleware erhält, an die Postdienststelle. Außerdem leitet er alle Transportbeschreibungen, die er über die Eingangsschnittstelle `Benachrichtbar` bekommt, mit Hilfe einer Nachrichtenvermittlung an den entsprechenden Postzustelldienst-Client weiter.

Die Umgangsformen (siehe Abbildung 36) des Postzustelldienst-Servers sind identisch mit denen der Postdienststelle und werden daher an dieser Stelle nicht erörtert. Statt dessen verweise ich auf den entsprechenden Abschnitt 5.3.3.

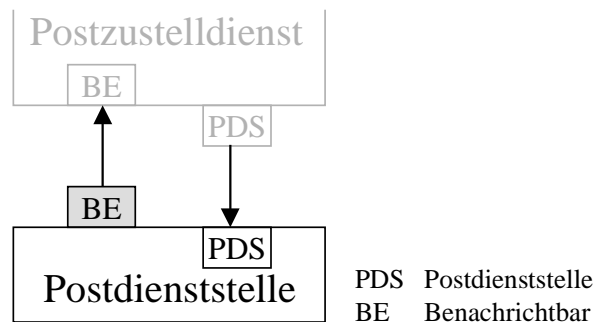


**Abbildung 36: Umgangsformen des Postzustelldienst-Servers**

### 5.3.3 Postdienststelle

Der Entwurf aus Abschnitt 5.1 sieht eine Postdienststelle vor, auf die zentral zugegriffen werden kann. Zusammen mit dem Postzustelldienst erbringt sie die Dienstleistung, Materialien arbeitsplatzübergreifend auszutauschen.

Um mit dem Postzustelldienst kommunizieren zu können, verfügt die Postdienststelle über eine gleichnamige Eingangsschnittstelle und die Ausgangsschnittstelle **Benachrichtbar** (siehe Abbildung 37).



**Abbildung 37: Ein- und Ausgangsschnittstellen der Postdienststelle**

Der Postzustelldienst nimmt die Dienstleistungen der Postdienststelle über die Eingangsschnittstelle **Postdienststelle** in Anspruch. Während in umgekehrter Richtung die Postdienststelle über die Ausgangsschnittstelle **Benachrichtbar** die Benachrichtigung der Empfänger an den Postzustelldienst gibt.

Die Postdienststelle hat mehrere Aufgaben zu erfüllen. So ist sie für die Verwaltung aller gültigen Empfängeradressen zuständig. Der Grund dafür ist, daß die Postdienststelle alle Materialien, die transportiert werden sollen, solange zwischenspeichert, bis der Empfänger sie bei der Postdienststelle anfordert. Ohne diesen Verwaltungsaufwand wäre es für die Postdienststelle ein unlösbares Problem, zu entscheiden, ob ein zwischengespeichertes Material nicht vom Empfänger abgeholt wird, weil er gerade nicht erreichbar ist, oder weil es

diesen Empfänger überhaupt nicht gibt. Auch ein Zeitlimit, innerhalb dessen ein Material bei der Postdienststelle abgeholt sein muß, ist keine gute Lösung für das Problem. Schließlich kann der Empfänger sich im Urlaub befinden und wäre dann zu einem späteren Zeitpunkt wieder erreichbar.

Eine weitere Aufgabe ist die Verwaltung der zuzustellenden Materialien. So muß die Postdienststelle prüfen können, ob ein Material zustellbar ist. Wenn ja, dann nimmt sie das Material an und speichert es solange zwischen, bis der Empfänger es anfordert. Wenn ein Material angefordert wird, prüft sie, ob das Material vorhanden ist und händigt es aus.

Neben der Verwaltung der Materialien gehört es auch zur Aufgabe, den Empfänger über den Neuzugang eines Materials in der Postdienststelle zu benachrichtigen. Damit der Empfänger auch darüber informiert wird, was für ein Material für ihn in der Postdienststelle gelagert wird, besteht die Benachrichtigung aus der Transportbeschreibung (siehe Abschnitt 5.2.2).

In der Abbildung 38 sind die Umgangsformen dargestellt, welche die genannten Aufgaben widerspiegeln. Im folgenden gehe ich im einzelnen darauf ein.

```
Postdienststelle
-----
sende (TransportierbaresMaterial)
hole Material (Transportbeschreibung)
ist sendbar (Transportbeschreibung)
Inhaltsverzeichnis (Transportadresse)
Adreßliste ( )
lösche Adresse (Transportadresse)
registriere Adresse (Transportadresse)

an-/abmelden (Benachrichtbar)
```

### Abbildung 38: Umgangsformen der Postdienststelle

Die erste Umgangsform bietet die Möglichkeit, ein Material zu versenden. In diesem Fall nimmt die Postdienststelle das Material entgegen und speichert es zwischen. Anschließend sendet sie die Transportbeschreibung des Materials als Benachrichtigung für den Empfänger über die Ausgangsschnittstelle `Benachrichtbar`. Als Vorbedingung für diese Umgangsform gilt, daß das Material versendbar sein muß. Falls die Empfängeradresse ungültig ist, wird der Absender benachrichtigt.

Weiterhin kann ein Material anhand einer konkreten Transportbeschreibung abgeholt werden. Jedoch ist nicht garantiert, daß das Material auch ausgehändigt wird. Es ist nämlich durchaus möglich, daß es unmittelbar zuvor über den Postzustelldienst an einen anderen Arbeitsplatz ausgeliefert wurde, da sich mehrere Arbeitsplätze für eine Empfängeradresse interessieren können. Dieser Fall der Mehrdeutigkeit zwischen Empfängeradresse und Arbeitsplatz wurde bereits bei der Konzeption erörtert (siehe Abschnitt 5.2). Kann jedoch das Material geliefert werden, dann befindet es sich automatisch nicht mehr in der Postdienststelle. Diese Vorgehensweise entspricht der Forderung nach Einheit von Ort und Zeit der Materialien (siehe Abschnitt 4), da ansonsten das selbe Material erneut angefordert werden könnte.

Die Postdienststelle bietet die Möglichkeit zu erfragen, ob ein transportierbares Material versendbar ist. Hierbei wird geprüft, ob eine Empfänger- und eine Absenderadresse beim Material gesetzt wurden.

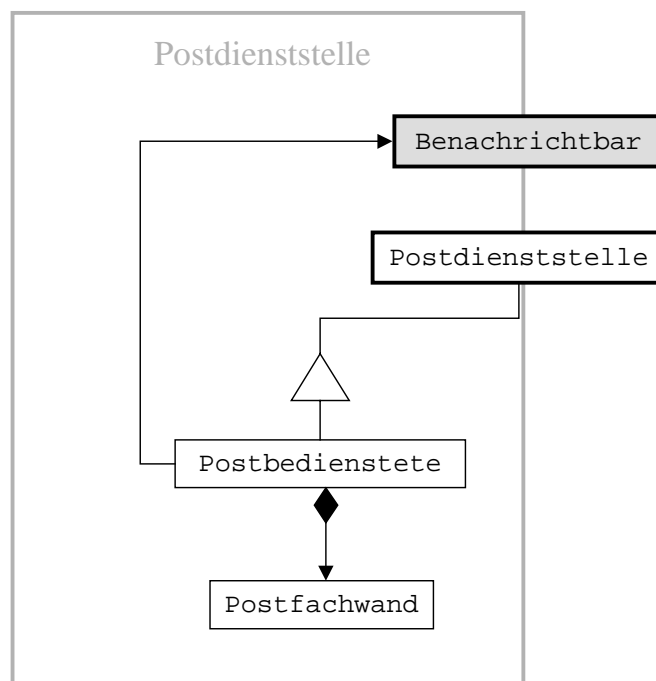
Unter Angabe einer Transportadresse (siehe Abschnitt 5.2.1) kann das Inhaltsverzeichnis aller zu einer Empfängeradresse gespeicherten Materialien erfragt werden. Dieses Inhaltsverzeichnis setzt sich aus den Transportbeschreibungen (siehe Abschnitt 5.2.2) der Materialien zusammen.

Neben den bisher genannten Umgangsformen, gibt es noch solche, die für die Adreßverwaltung benötigt werden. Um zu erfahren, welche Adressen gegenwärtig gültig sind, kann die Postdienststelle nach einer Adreßliste befragt werden. Da jederzeit neue Adressen hinzukommen können bzw. wieder gelöscht werden, hat die Adreßliste nur informatorischen Charakter und widerspiegelt den Zustand im Moment der Abfrage.

Weiterhin bietet die Postdienststelle die Möglichkeit, eine gültige Adresse zu löschen. Das bedeutet, daß dann keine Materialien mehr an diese Adresse gesendet werden können. Der Löschvorgang der Adresse bewirkt außerdem, daß auch alle Materialien, die zu dieser Empfängeradresse zwischengespeichert sind, unwiderruflich vernichtet werden. Um das zu verhindern, kann zuvor ein Inhaltsverzeichnis zu einer Adresse erfragt werden, um sich dann alle noch gespeicherten Materialien herausgeben zu lassen.

Falls eine neue Empfängeradresse registriert werden soll, so bietet die Postdienststelle auch dafür eine Umgangsform an. Nachdem die Registrierung einer neuen Adresse erfolgt ist, können Materialien an diese gesendet werden.

Die letzte Umgangsform dient dazu, daß der Postzustelldienst, der die Benachrichtigungen der Postdienststelle zu den jeweiligen Handhabungskomponenten überträgt, an- und abgemeldet werden kann.



**Abbildung 39: Aufbau der Postdienststelle**

Die Abbildung 39 stellt den internen Aufbau der Postdienststelle dar. Ich habe diesen Aufbau gewählt, da so die Funktionsweise der Postdienststelle sehr gut veranschaulicht wird. Dieser Aufbau ist jedoch nicht zwingend, solange die oben beschriebenen Ein- und Ausgangsschnittstellen erfüllt werden.

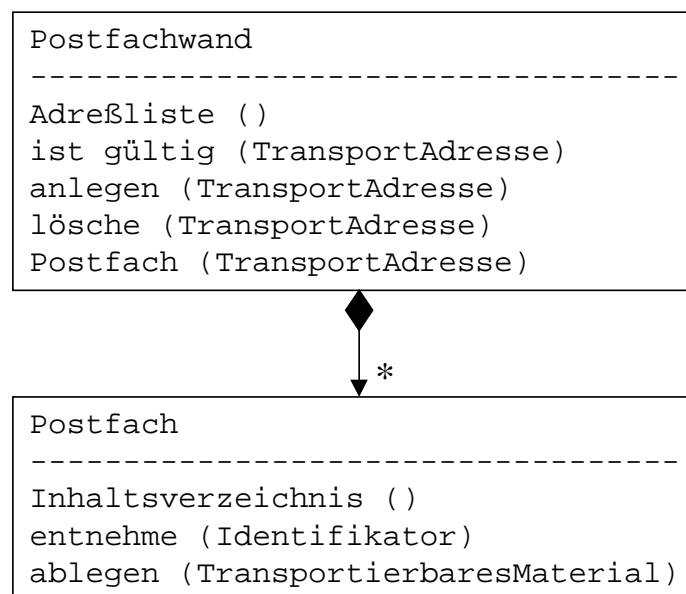
In der Abbildung wird deutlich, daß der Postbedienstete die Schnittstelle Postdienststelle erfüllt. Er realisiert die über die Schnittstelle angebotenen Dienstleistungen und nutzt eine Postfachwand zum Zwischenspeichern der zu transportierenden Materialien.

In den folgenden Abschnitten gehe ich detaillierter auf die einzelnen Bestandteile der Postdienststelle ein.

### 5.3.3.1 Postfachwand

Die Postfachwand ist ein Bestandteil der anwendungsfachlich motivierten Realisierung der Postdienststelle. Sie wird ausschließlich nur vom Postbediensteten verwendet und dient der Zwischenlagerung der zu transportierenden Materialien in der Postdienststelle.

Der Aufbau der Postfachwand ist in Abbildung 40 ersichtlich. Sie setzt sich aus einer Menge von Postfächern zusammen, wobei es für jede Empfängeradresse genau ein Postfach gibt. Dadurch können die zu transportierenden Materialien eindeutig den Empfängeradressen zugeordnet werden.



**Abbildung 40: Aufbau und Umgangsformen der Postfachwand mit Postfächern**

Über die von der Postfachwand angebotenen Umgangsformen kann eine Menge von Postfächer verwaltet und zugegriffen werden. So besteht die Möglichkeit, eine Adreßliste zu erfragen. Diese Adreßliste enthält alle Adressen, für die es Postfächer gibt. Weiterhin kann sondiert werden, ob es zu einer bestimmten Adresse ein Postfach in der Postfachwand gibt. Sollte dem nicht so sein, so kann ein neues Postfach zu einer Adresse angelegt werden. Wenn ein Postfach nicht mehr benötigt wird, so kann es wieder gelöscht werden. Dabei wird auch

der Inhalt des Postfaches mit zerstört. Weiterhin besteht die Möglichkeit, das zu einer Adresse gehörende Postfach zu erfragen.

Das Postfach wiederum hat seine eigenen Umgangsformen. So kann das Inhaltsverzeichnis erfragt werden, das sich aus den Transportbeschreibungen der in ihm abgelegten Materialien zusammensetzt. Weiterhin gestattet es den Zugriff auf einzelne Materialien, indem ein Identifikator des Materials angegeben wird. Der Identifikator ist Bestandteil der Transportbeschreibungen aus dem erfragbaren Inhaltsverzeichnis. Wurde ein Material aus dem Postfach entnommen, dann befindet es sich anschließend nicht mehr darin, um die Einheit von Ort und Zeit der Materialien zu gewährleisten. Weiterhin ist es möglich, ein transportierbares Material in das Postfach abzulegen.

### **5.3.3.2 Postbedienstete**

Der Postbedienstete ist Bestandteil der Postdienststelle. Zusammen mit der Postfachwand erbringt er die Dienstleistungen, die über die Schnittstelle Postdienststelle angeboten werden.

Der Aufbau des Postbediensteten ist in Abbildung 39 ersichtlich. Da er lediglich die bereits beschriebenen Umgangsformen der Postdienststelle realisiert und somit auch die gleichen Aufgaben wahrnimmt, gehe ich im folgenden nicht näher auf die Umgangsformen ein. Vielmehr beschreibe ich im folgenden, wie die Umgangsformen mit Hilfe der Postfachwand erbracht werden.

Wenn beim Postbediensteten die Adreßliste erfragt wird, so gibt er die der Postfachwand zurück. Der Grund dafür ist, daß nur solche Adressen gültig sind, für die es Postfächer in der Postfachwand gibt.

Wird der Postbedienstete beauftragt, eine Adresse zu löschen, dann bedeutet es, daß er das entsprechende Postfach aus der Postfachwand entfernt. Sollte zu einem späteren Zeitpunkt versucht werden, ein Material an eine gelöschte Adresse zu senden, so findet der Postbedienstete kein entsprechendes Postfach für die Zwischenlagerung mehr vor. In diesem Fall legt er das Material in das Postfach des Absenders ab.

Wenn beim Postbediensteten eine neue Adresse registriert wird, so sorgt er dafür, daß ein entsprechendes Postfach in der Postfachwand neu angelegt wird. Dadurch ist er anschließend in der Lage, alle Materialien für diese Adresse zwischenzuspeichern und somit für eine Zustellung zu sorgen.

Soll ein Material versendet werden, so legt der Postbedienstete es in das Postfach des Empfängers ab. Gibt es für die Empfängeradresse kein entsprechendes Postfach, so legt er es in das des Absenders. Anschließend übermittelt er die Transportbeschreibung als Benachrichtigung über die Ausgangsschnittstelle Benachrichtbar. Wenn ein Material in ein Postfach abgelegt wurde, so verbleibt es dort solange, bis es wieder entnommen wird.

Wenn das Inhaltsverzeichnis aller zu einer Empfängeradresse zwischengelagerten Materialien angefordert wird, so liefert der Postbedienstete das Inhaltsverzeichnis des entsprechenden Postfaches. Das ist möglich, weil es eine umkehrbar eindeutige Abbildung der Transportadressen auf Postfächer in der Postfachwand gibt.

Wird ein Material anhand einer Transportbeschreibung erfragt, so läßt sich der Postbedienstete das Postfach aus der Postfachwand zu der Empfängeradresse aus der



Transportbeschreibung geben. Aus diesem Postfach entnimmt er dann das Material, das den Identifikator aus der Transportbeschreibung hat. Anschließend gibt er das so erhaltene Material als Ergebnis der Anfrage zurück.

Die Umgangsform, um zu prüfen, ob ein Material versendbar ist, ist ohne die Verwendung der Postfachwand realisiert. Der Postbedienstete braucht nur zu sondieren, ob die Empfänger- und die Absenderadresse am Material gesetzt sind.

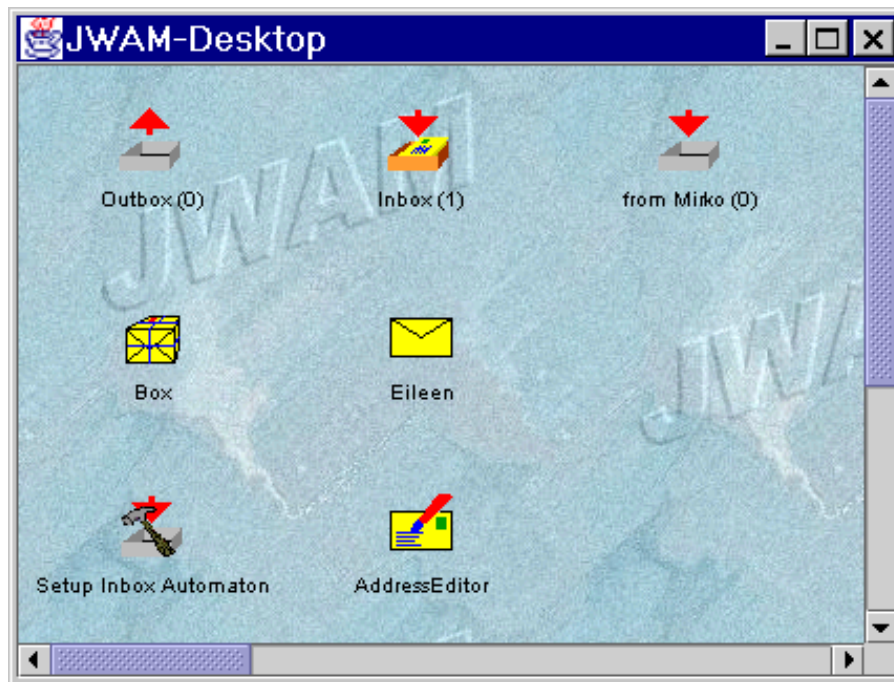
Das An- und Abmelden der Komponente, die sich für die Benachrichtigungen interessiert, geschieht ebenfalls unabhängig der Postfachwand. Bei dieser Umgangsform speichert der Postbedienstete lediglich eine Referenz auf die angemeldete Komponente bzw. löscht sie wieder. Diese Referenz verwendet der Postbedienstete, wenn er über die Ausgangsschnittstelle `Benachrichtbar` Benachrichtigungen in Form von Transportbeschreibungen absendet.

#### **5.4 Werkzeugunterstützung**

In den folgenden Abschnitten werde ich die Werkzeuge, insbesondere deren Handhabung, beschreiben. Sie unterstützen den Benutzer bzw. den Administrator beim Umgang mit dem Postversandsystem und dienen somit der Erfüllung der geforderten Benutzungsmodele aus Abschnitt 4.3.

Ich möchte betonen, daß das Postversandsystem keine allgemeingültige Endfassung für alle Anwendungskontexte ist. Vielmehr stellt es eine Basisunterstützung zur Verfügung, die je nach Anwendungskontext angepaßt werden kann. Aus diesem Grund haben die im folgenden vorgestellten Werkzeuge nur beispielhaften prototypischen Charakter. Sie können je nach Anwendungskontext optimiert werden, sowohl in Fragen des Erscheinungsbildes, der Benennung als auch deren Handhabung.

Doch bevor ich auf die Werkzeuge im einzelnen eingehe, möchte ich noch beispielhaft einen mit dem Postversandsystem ausgestatteten Arbeitsplatz vorstellen, wie ihn ein Benutzer vorfinden könnte. Die Abbildung 41 zeigt den Arbeitsplatz visualisiert durch den JWAM-Desktop.



**Abbildung 41: Der Arbeitsplatz**

Auf dem Desktop sind die Icons der verschiedenen Dinge zu sehen, die ich im folgenden kurz nennen werde (von oben links nach rechts beginnend):

1. *Leerer Postausgangskorb*

Am Icon und am Namen läßt sich erkennen, daß es sich um einen leeren Postausgangskorb handelt.

2. *Mit einem Material befüllter Standardposteingangskorb*

Sowohl das Icon als auch der Name signalisieren, daß es sich um einen gefüllten Posteingangskorb handelt. Im Namen ist auch die genaue Anzahl der Materialien enthalten, die sich im Postkorb befinden. Weiterhin verrät der Name, daß es sich hierbei um einen Standardposteingangskorb handelt.

3. *Für den Absender "Mirko" reservierter und leerer Posteingangskorb*

Sowohl das Icon als auch der Name zeigen an, daß es sich um einen leeren Posteingangskorb handelt. Der Name läßt zusätzlich noch erkennen, daß in ihm nur Materialien vom Absender "Mirko" abgelegt werden.

4. *Ein Transportbehälter*

In diesen transportierbaren Behälter können Materialien abgelegt werden, die ortsveränderbar sind. Dadurch hat der Benutzer die Möglichkeit, auch ortsveränderbare Materialien mit dem Postversandssystem zu versenden.

5. *Ein transportierbares Material*

Dieses Material ist nicht nur transportierbar sondern auch mit dem Adreßeditor adressierbar. Es hat keine weitere fachliche Bedeutung und dient lediglich zum Testen des Postversandsystems. Der Name des Materials verrät, wer der Empfänger ist. Das Icon deutet an, daß dieses Material mit dem Postversandssystem versendet werden kann.

Sowohl der Name als auch das Icon können je nach Anwendungsfall modifiziert werden und haben in der Abbildung nur beispielhaften Charakter.

#### 6. *Einstellwerkzeug des Posteingangsautomaten*

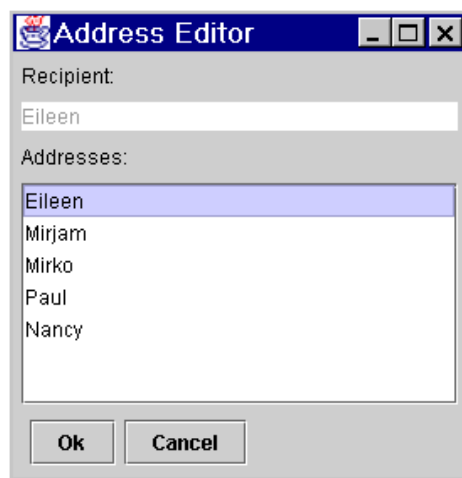
Dieses Werkzeug dient zur Konfiguration des Posteingangsautomaten. Es ist in Abschnitt 5.4.4 detailliert beschrieben.

#### 7. *Adreßeditor*

Mit dem Adreßeditor kann ein adressierbares Material (siehe Abschnitt 5.2.2) mit einer Empfängeradresse versehen werden. Dieses Werkzeug ist in Abschnitt 5.4.1 näher erläutert.

### 5.4.1 Adreßeditor

Voraussetzung für die korrekte Zustellung eines transportierbaren Materials ist, daß an diesem eine Empfängeradresse gesetzt ist. Aus diesem Grund ist in den geforderten Benutzungsmodellen (siehe Abschnitt 4.3) der Adreßeditor vorgesehen. Er dient dem Benutzer zum Adressieren eines Materials mit der Schnittstelle `AdressierbaresMaterial` (siehe Abschnitt 5.2.2). In der Abbildung 42 ist dargestellt, wie er sich dem Benutzer präsentiert.



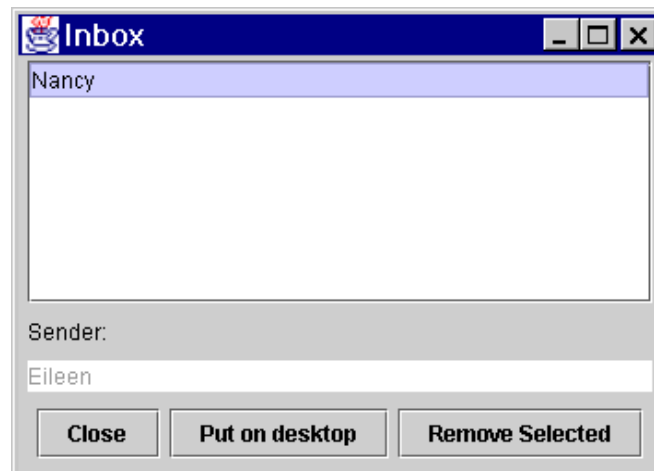
**Abbildung 42: Der Adreßeditor**

Dieses Werkzeug wird zu einem adressierbaren Material gestartet. Es zeigt dem Benutzer die aktuelle Empfängeradresse des Materials sowie eine Liste aller aktuell gültigen Adressen. Der Benutzer kann eine Adresse aus der Liste selektieren. Dieser Eintrag wird ihm sofort als die neue Empfängeradresse angezeigt. Wenn der Benutzer seine Eingabe mit dem Ok-Knopf bestätigt, schließt sich das Werkzeug und setzt am Material die neue Empfängeradresse.

### 5.4.2 Posteingangskorbeditor

Im erweiterten Benutzungsmodell (siehe Abschnitt 4.3.2) ist ein spezielles Werkzeug für Posteingangskörbe vorgesehen, das neben dem Inhalt auch die Absenderadresse des

jeweiligen empfangenen Materials anzeigt. Der Posteingangskorbeditor kommt dieser Forderung nach (siehe Abbildung 43).



**Abbildung 43: Der Posteingangskorbeditor**

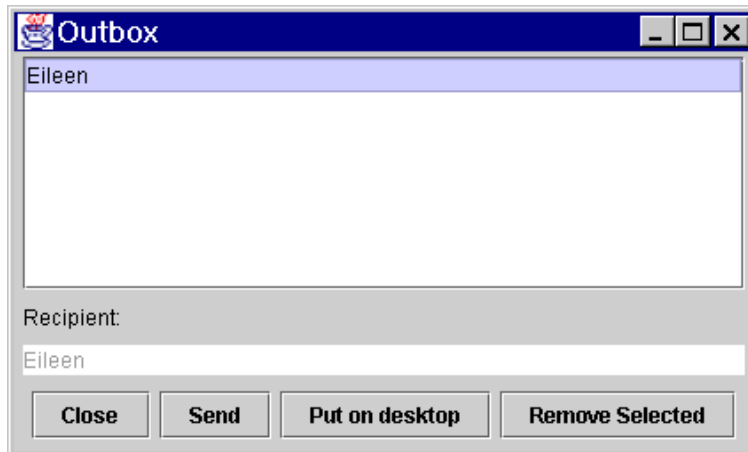
Das Werkzeug wird zu einem Posteingangskorb gestartet und zeigt dem Benutzer eine Liste aller darin befindlichen Materialien sowie ein Feld für die Absenderadresse an. Wenn der Benutzer ein Material in der Liste selektiert, so wird ihm der entsprechende Absender in dem dafür vorgesehenen Feld eingeblendet. Weiterhin hat er die Option, mit Hilfe der zur Verfügung gestellten Knöpfe, das selektierte Material zu löschen oder auf den Desktop zu verschieben.

### 5.4.3 Postausgangskorbeditor

Der erweiterte Umgang mit dem Postversandsystem (siehe Abschnitt 4.3.2) sieht für den Benutzer ein Werkzeug vor, mit dem er manuell den Versandevorgang starten kann. Damit wird dem Benutzer die Möglichkeit eingeräumt, den Sendezeitpunkt der Materialien zu beeinflussen.

Der Postausgangskorbeditor erfüllt diese Anforderung. Mit ihm kann der Benutzer seinen Postausgangskorb verwalten und die Versendung von Materialien manuell starten (siehe Abbildung 44).

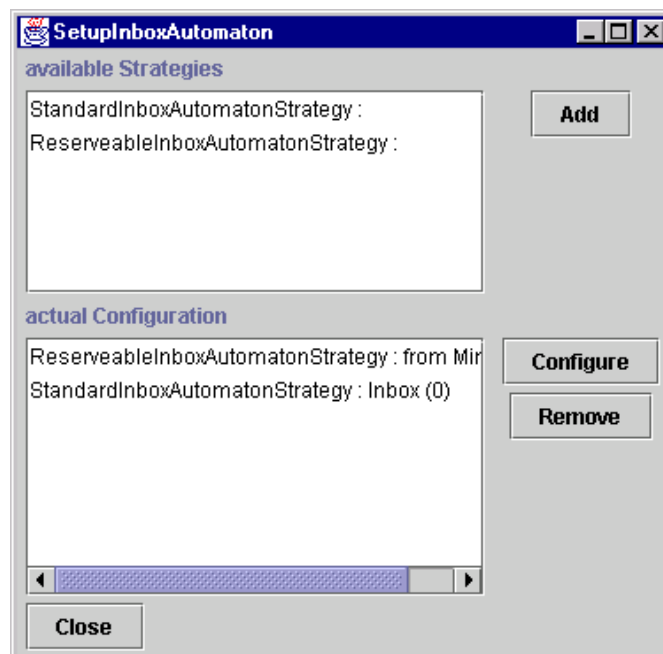
Ähnlich wie beim Posteingangskorbeditor wird das Werkzeug zu einem Postausgangskorb gestartet. Es zeigt dem Benutzer eine Liste aller in dem Postkorb befindlichen Materialien an. Weiterhin gibt es ein Textfeld, in dem die Empfängeradresse des in der Liste selektierten Materials angezeigt wird. Dem Benutzer stehen verschiedene Knöpfe zur Verfügung, mit denen er ein selektiertes Material löschen oder auf den Desktop verschieben kann. Weiterhin hat er noch einen Knopf, mit dem er manuell die Versendung aller Materialien in dem Postausgangskorb starten kann. Wenn alle Materialien erfolgreich versendet werden konnten, ist die dargestellte Liste leer.



**Abbildung 44: Der Postausgangskorbeditor**

#### 5.4.4 Einstellwerkzeug des Posteingangsautomaten

Im Benutzungsmodell zur Konfiguration des Postversandsystems (siehe Abschnitt 4.3.3) ist für den Administrator ein Werkzeug vorgesehen, womit er Behälter als Posteingangskörbe einrichten kann. Das Einstellwerkzeug des Posteingangsautomaten leistet genau diese Unterstützung (siehe Abbildung 45).



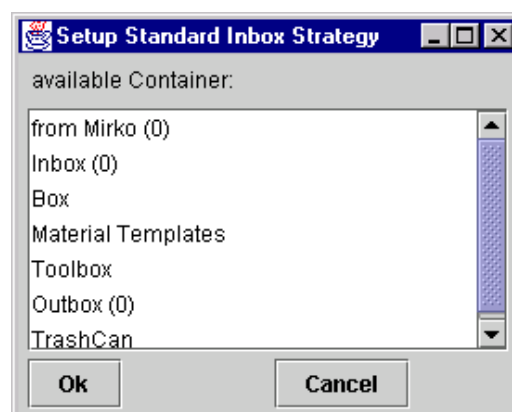
**Abbildung 45: Das Einstellwerkzeug des Posteingangsautomaten**

Es zeigt dem Administrator eine Auswahlliste aller zur Verfügung stehenden Posteingangsstrategien (siehe Abschnitt 5.3.1.2) an. Mit diesen Posteingangsstrategien kann der Administrator einen Behälter als einen bestimmten Posteingangskorb einrichten. Die zweite Liste, die angezeigt wird, enthält die aktuellen Posteingangsstrategien des

Posteingangsautomaten. Anhand der Einträge in dieser Liste läßt sich erkennen, welche Posteingangsstrategie mit welchem Behälter konfiguriert wurde.

Weiterhin werden verschiedene Knöpfe angeboten. Mit dem Add-Knopf wird eine in der ersten Liste ausgewählte Strategie dem Posteingangsautomaten hinzugefügt. Das erkennt der Administrator daran, daß ein neuer Eintrag in der zweiten Liste erscheint. Am Namen läßt sich jedoch erkennen, daß diese Strategie noch nicht eingerichtet wurde. Wenn der Administrator diesen Eintrag selektiert, kann er mit dem Configure-Knopf ein entsprechendes Einstellwerkzeug für die Posteingangsstrategie starten (siehe unten). Um eine Posteingangsstrategie des Posteingangsautomaten aus der zweiten Liste wieder zu löschen, braucht der Administrator diese nur zu selektieren und den Remove-Knopf zu drücken. Anschließend erscheint diese Posteingangsstrategie auch nicht mehr in der zweiten Liste. Wenn der Administrator mit der Einstellung des Posteingangsautomaten fertig ist, drückt er den Close-Knopf, um das Einstellwerkzeug zu schließen.

Wie bereits erwähnt, gibt es noch Einstellwerkzeuge für die jeweiligen Posteingangsstrategien. In Abbildung 46 ist das Einstellwerkzeug der Standardstrategie (siehe Abschnitt 5.3.1.2) dargestellt. Mit diesem kann der Standardstrategie ein Behälter zugeordnet werden, der dann als Standardposteingangskorb fungiert.

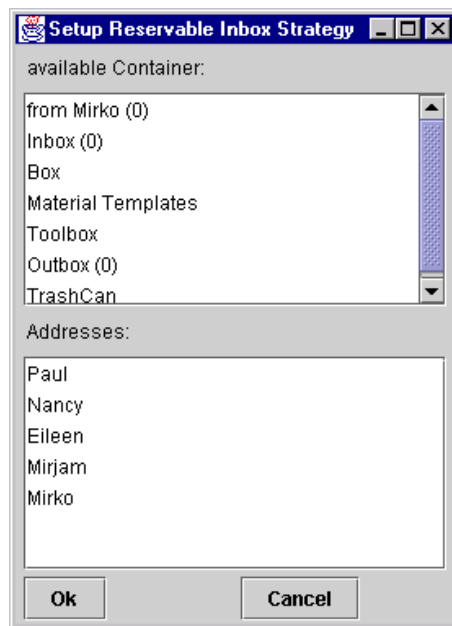


**Abbildung 46: Das Einstellwerkzeug der Standardstrategie**

Dieses Einstellwerkzeug zeigt eine Liste aller zur Verfügung stehenden Behälter an. Der Administrator kann einen Behälter selektieren und seine Auswahl mit dem Ok-Knopf bestätigen. Anschließend konfiguriert das Einstellwerkzeug die Standardstrategie mit diesem Behälter und schließt sich. Diese Änderung der Konfiguration der Standardstrategie ist anschließend in der zweiten Liste des Einstellwerkzeuges des Posteingangsautomaten (siehe Abbildung 45) am Namen der konfigurierten Strategie ersichtlich. Der Administrator hat jedoch auch die Möglichkeit, den Einstellvorgang mit dem Cancel-Knopf abubrechen. Dann schließt sich das Einstellwerkzeug, ohne die Einstellung der Standardstrategie zu ändern.

Das zweite Einstellwerkzeug ist für die Reservierungsstrategie (siehe Abschnitt 5.3.1.2). Es ist in Abbildung 47 dargestellt. Mit diesem kann der Reservierungsstrategie ein Behälter zugeordnet und ein Absender eingerichtet werden. Nach erfolgreicher Konfiguration mit diesem Werkzeug fungiert der Behälter als ein Posteingangskorb, in dem vom Postversandssystem nur Materialien mit dem eingerichteten Absender abgelegt werden.

Dieses Einstellwerkzeug zeigt eine Liste aller verfügbaren Behälter sowie eine Adreßliste an. Der Administrator kann der Reservierungsstrategie einen Behälter zuordnen, indem er einen aus der ersten Liste selektiert. Um die Reservierungsstrategie vollständig zu konfigurieren, muß er noch eine Absenderadresse setzen, indem er aus der zweiten Liste eine Adresse auswählt. Bestätigt der Administrator seine Auswahl mit dem Ok-Knopf, so wird die Reservierungsstrategie von diesem Einstellwerkzeug entsprechend eingerichtet. Anschließend schießt sich das Einstellwerkzeug. Bricht der Administrator jedoch den Einstellvorgang ab, indem er den Cancel-Knopf drückt, so bleibt die Konfiguration der Reservierungsstrategie unverändert.



**Abbildung 47: Das Einstellwerkzeug der Reservierungsstrategie**





## 6 Integration ins JWAM-Rahmenwerk

Das in dieser Arbeit vorgestellte Postversandsystem ist basierend auf dem JWAM-Rahmenwerk entwickelt und auch in diesem integriert. Aus diesem Grund stelle ich in diesem Kapitel den Aufbau des JWAM-Rahmenwerkes vor und zeige, wie das Postversandsystem integriert ist. Zu diesem Zweck unterteilt es sich in die folgenden Abschnitte:

Im Abschnitt 6.1 wird beschrieben, mit welchen Problemen Rahmenwerke konfrontiert sind. Als spezielle Lösung dieser Probleme stelle ich die WAM-Modellarchitektur vor.

Abschnitt 6.2 stellt die Softwarearchitektur des JWAM-Rahmenwerkes vor, die sich an der WAM-Modellarchitektur orientiert. Dabei beschränke ich mich auf ausgewählte Bestandteile des Rahmenwerkes und nenne gegebenenfalls, ob und wie sie vom Postversandsystem verwendet werden. Anschließend wird gezeigt, an welcher Stelle des JWAM-Rahmenwerkes das Postversandsystem angesiedelt ist.

### 6.1 WAM-Modellarchitektur

In der heutigen Zeit sieht sich der Anwendungsentwickler dem Spannungsfeld zwischen der Forderung nach Individualsoftware und effizienten und kostengünstigen Entwicklungsprozessen ausgesetzt. Ein Ausweg aus diesem Spannungsfeld sind Rahmenwerke. Sie stellen eine generische Lösung für die Anwendungsentwicklung dar. Das Problem ist, daß ein Rahmenwerk nur die Anwendungsentwicklung für eine fachlich beschränkte Domäne wirkungsvoll unterstützen kann.

Ein weiteres Problem ist, daß die Anwendungskontexte immer komplexer werden, für die Software entwickelt werden soll. Folglich müssen auch die Rahmenwerke immer komplexer werden, um auch diese Anwendungskontexte angemessenen zu unterstützen. Aufgrund der Komplexität kann es sehr schnell passieren, daß das Rahmenwerk unübersichtlich wird und somit dem Anwendungsentwickler verborgen bleibt, was das Rahmenwerk zu leisten fähig ist.

Neben den bereits genannten Problemen lassen sich noch zwei weitere identifizieren, die bei der Verwendung von Rahmenwerken auftreten können. So sollte zwar ein Rahmenwerk während der Anwendungsentwicklung nicht modifiziert werden, jedoch kann es notwendig werden, daß ein Rahmenwerk an geänderte Umstände angepaßt werden muß. Solche Änderungen wirken sich fast zwangsläufig auch auf die mit dem Rahmenwerk entwickelten Anwendungssysteme aus. Ein geschickter Aufbau des Rahmenwerkes könnte die Tragweite von Änderungen am Anwendungssystem minimieren.

Wenn der Anwendungsentwickler ein komplexes Rahmenwerk einsetzen möchte, kann er es schwer haben, Entwurfsentscheidungen zu lokalisieren. Somit würde ihm die nötige Orientierung fehlen, um zu wissen, wie er das Rahmenwerk verwenden soll. Optimal wäre es, wenn er einfach und schnell feststellen kann, welche Entwurfsentscheidungen im Rahmenwerk getroffen wurden und an welchen Stellen er neue Funktionalität ansiedeln kann.

Um den genannten Problemen zu begegnen, bietet es sich an, das Rahmenwerk so zu organisieren, daß es überschaubar und handhabbar wird. Das Rahmenwerk sollte also gröbere Strukturen aufweisen und nicht auf der Ebene der einzelne Klassen betrachtet werden müssen.

Wenn sich diese Strukturen an fachlichen Abstraktionskriterien orientieren (vgl. [BGK+97]), wird im Kontext des WAM-Ansatzes von einer *Softwarearchitektur* gesprochen. Die Strukturen stellen die Komponenten der Softwarearchitektur dar. Solche Komponenten können Klassenbibliotheken, Black-box- und White-box-Rahmenwerke sein.

**Softwarearchitektur:**

"Eine Softwarearchitektur bezeichnet die Modelle und die konkreten Komponenten eines Softwaresystems in ihrem statischen und dynamischen Zusammenspiel. Sie kann selbst als explizites Modell dargestellt werden.

Eine Softwarearchitektur beschreibt ein konkretes System in seinem Anwendungskontext."  
([Zül98], S. 324)

Weiterhin muß festgehalten werden, daß die Komponenten einer Softwarearchitektur über Verbindungsstücke gekoppelt werden. Diese Verbindungsstücke legen auf der Ebene von konkreten Schnittstellen fest, wie die Komponenten miteinander interagieren. Als Verbindungsstücke kommen Entwurfsmuster, Vererbung und Benutzt-Beziehungen in Frage.

Bei der Entwicklung von Softwarearchitekturen können neben den Komponenten noch abstraktere Regeln hinzugezogen werden, die beschreiben, wie die Softwarearchitektur gestaltet und konstruiert werden kann. Diese Regeln dienen dem Architekturstil (vgl. [Bäu98]) oder, wie es im Kontext des WAM-Ansatzes genannt wird, der *Modellarchitektur*. Eine solche Modellarchitektur gibt sozusagen einen Rahmen für die Konstruktion von Softwarearchitekturen vor.

**Modellarchitektur:**

"Eine Modellarchitektur beschreibt die allgemeinen Prinzipien hinter einer Softwarearchitektur. Sie umfaßt die grundlegenden Elemente, deren Verknüpfungen und die Regeln, die für eine Softwarearchitektur gelten.

Eine Modellarchitektur gibt Anleitung bei der softwaretechnischen Realisierung eines Softwaresystems." ([Zül98], S. 325)

Die WAM-Modellarchitektur sieht vor, die Softwarearchitektur von Rahmenwerken in Schichten zu strukturieren (vgl. [GLL+99], [Zül98]). Dabei wird zwischen folgenden Arten von Schichtenarchitekturen unterschieden:

1. *Protokollbasiert*

Bei einer protokollbasierten Schichtenarchitektur sind die Schichten hierarchisch angeordnet. Dabei liegt die Schicht, welche die Basisdienste erbringt am tiefsten. Auf dieser Schicht setzen immer anwendungsnähere Schichten auf. Die oberste Schicht ist somit am anwendungsorientiertesten.

Weiterhin verbirgt jede Schicht hinter ihrer Schnittstelle, die sie einer höheren anbietet, wie sie implementiert ist und auf welchen Schichten sie wiederum aufsetzt. Mit diesem Geheimnisprinzip wird das Ziel verfolgt, daß eine Schicht immer nur auf die Schnittstelle der nächst niederen zugreifen kann. Dadurch wirken sich Änderungen in der Implementation immer nur lokal auf die jeweilige Schicht aus. Auch die Tragweite einer Schnittstellenänderung ist sehr begrenzt, da sie lediglich die höher liegende Schicht betrifft.

Die Schichten einer protokollbasierten Schichtenarchitektur werden meistens mit Modulen realisiert, da sie als Protokoll eine Schnittstelle anbieten, hinter der sie die Implementation verbergen. Damit haben sie die beschriebenen Eigenschaften einer protokollbasierten Schicht. Ein Beispiel für eine protokollbasierte Schichtenarchitektur ist das OSI-Referenzmodell (vgl. [Ker93]).

## 2. *Objektorientiert*

Bei der objektorientierten Schichtenarchitektur ist nicht nur die reine Benutzt-Beziehung möglich, wie bei der protokollbasierten, sondern auch die Vererbung. Die Schichten sind hierarchisch angeordnet und die höheren sind wieder anwendungsnäher als die niederen. Dabei bedeutet das im Falle der Vererbung, daß eine höhere Schicht eine niedrigere spezialisiert bzw. konkretisiert. Es gilt das Prinzip, daß immer nur innerhalb einer Schicht bzw. von einer niederen zu einer höheren vererbt wird.

Die Benutzt-Beziehung ist auch bei der objektorientierten Schichtenarchitektur von oben nach unten gerichtet, d.h. eine Schicht kann immer nur eine niedrigere benutzen.

Zur Realisierung einer objektorientierten Schichtenarchitektur können Klassenbibliotheken als auch Rahmenwerke dienen. Dabei hat eine Schicht die Bedeutung, daß sie die Elemente der Softwarearchitektur strukturell zu einer Einheit zusammenfaßt. Sie stellt selbst keine eigenständige softwaretechnische Komponente dar. Somit kann eine Schicht auch keine eigene Schnittstelle anbieten. Vielmehr ist ihre Schnittstelle die Menge der Schnittstellen ihrer Komponenten. Das steht im Gegensatz zu der protokollbasierten Schichtenarchitektur, bei der jede Schicht als eigenständige softwaretechnische Komponente mit ihrer eigenen Schnittstelle realisiert wird.

Hervorzuheben ist, daß die WAM-Modellarchitektur weder eine rein protokollbasierte noch eine objektorientierte Schichtenarchitektur vorsieht. Vielmehr können die Schichten sowohl protokollbasiert als auch objektorientiert sein, wie in der folgenden Definition einer Schicht in der WAM-Modellarchitektur deutlich wird.

### **Schicht in der WAM-Modellarchitektur:**

"Eine Schicht organisiert die softwaretechnischen Komponenten einer Modellarchitektur zu einer fachlich und technisch motivierten Entwurfs- und Konstruktionseinheit. Eine Schicht hat selbst keine Schnittstelle und keine Beziehung zu anderen Schichten; Schnittstellen und Beziehungen über Verbindungsstücke haben nur die enthaltenen Komponenten.

Die verschiedenen Schichten einer Modellarchitektur sind hierarchisch aufgebaut. Je nach Sichtbarkeit und Verwendungszusammenhang sprechen wir von einer protokollbasierten oder objektorientierten Schicht.

Als Komponenten innerhalb einer Schicht verwenden wir Klassenbibliotheken und Rahmenwerke. Die Verbindungsstücke sind durch Konstruktionsmuster, Benutzt-Beziehung oder Vererbung realisiert." ([Zül98], S. 352)

Die WAM-Modellarchitektur besteht aus den folgenden drei Schichten (siehe Abbildung 48):

### 1. *Systembasisschicht*

Die Systembasisschicht dient im wesentlichen dazu, technische Komponenten zu kapseln. Solche Komponenten können z.B. bereits existierende Systeme sein, wie ein Fenster-

oder Betriebssystem. Aber auch andere Komponenten werden in dieser Schicht untergebracht, die technische Problemlösungen darstellen, wie z.B. eine Speicherbereinigung oder persistente Datenspeicher.

Um die genannten technischen Komponenten zu kapseln, werden Black-box-Rahmenwerke und Klassenbibliotheken angeboten. Diese Schicht ist somit als eine rein protokollbasierte Schicht gedacht. Das ist insbesondere daher von großer Bedeutung, weil an der Schnittstelle dieser Schicht nicht erkennbar sein darf, wie sie realisiert ist.

Da diese Schicht nur technische Komponenten kapselt, ist sie vollkommen unabhängig vom Anwendungskontext. Das bedeutet, daß sie als Grundlage in den verschiedensten Anwendungssystemen verwendet werden kann.

## 2. *Technologieschicht*

Die Technologieschicht setzt auf der Systembasisschicht auf. Sie ist abstrakter als die Systembasisschicht und enthält die Modelle der verwendeten Technik. Damit stellt die Technologieschicht ein Niveau zur Verfügung, daß weitgehend losgelöst von der verwendeten Technik ist. So könnte sie z.B. den Umgang mit einem relationalen Datenmodell unterstützen, anstatt "nur" den Zugriff auf eine bestimmte relationale Datenbank anzubieten.

Die Technologieschicht ist als eine objektorientierte Schicht vorgesehen. Sie enthält im wesentlichen White-box-Rahmenwerke und kann um einzelne Black-box-Rahmenwerke für vorgefertigte Standardlösungen erweitert werden.

Da diese Schicht keinen bestimmten Anwendungsbereich fokussiert, ist sie anwendungsneutral. Somit läßt sich die Technologieschicht in den verschiedensten Anwendungsbereichen wiederverwenden, wenngleich die angebotenen Modelle die potentiell möglichen Anwendungsbereiche etwas einschränken.

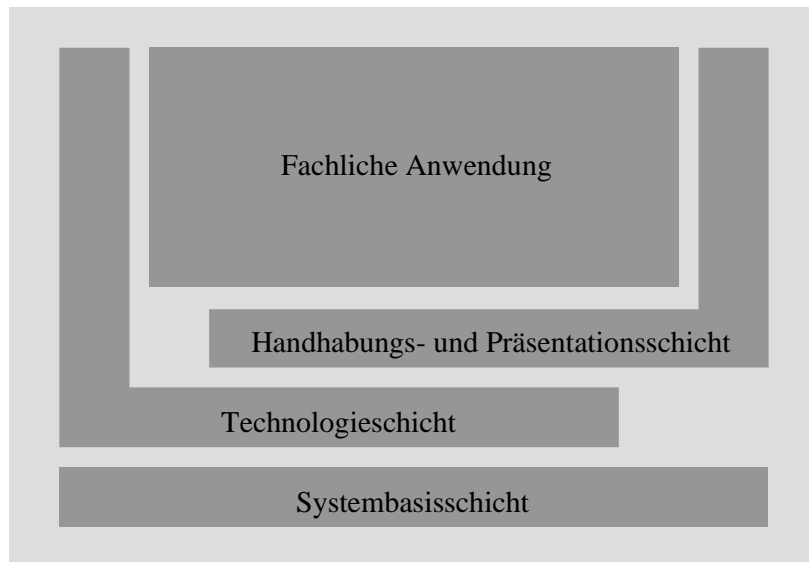
## 3. *Handhabungs- und Präsentationsschicht*

Die Handhabungs- und Präsentationsschicht verwendet die Technologie- und die Systembasisschicht. Sie enthält die Komponenten und Entwurfsentscheidungen, welche die Handhabung und Präsentation interaktiver Anwendungssoftware betreffen. Dabei orientieren sich die Handhabung und die Präsentation am Leitbild und den zugehörigen Entwurfsmetaphern des WAM-Ansatzes (siehe Kapitel 3). Zu den Komponenten und Entwurfsentscheidungen dieser Schicht zählen nicht nur die, welche die Konstruktion von Anwendungen betreffen, sondern auch solche, die den Umgang ausgestalten. Ich meine damit sowohl vorgefertigte Standardimplementationen, wie z.B. die Entwurfsmuster für den Bau von Werkzeugen, Materialien und Automaten, als auch vollständig implementierte Komponenten, die den Arbeitsplatz des Benutzers ausgestalten, wie z.B. Behälter, Auflister und der Papierkorb.

Um die in dieser Schicht u.a. enthaltenen Entwurfsentscheidungen dem Anwendungsentwickler zugänglich zu machen, werden diese in White-box-Rahmenwerken angeboten. Daraus resultiert, daß die Handhabungs- und Präsentationsschicht objektorientiert ist.

Da in dieser Schicht keine anwendungsspezifischen Entwurfsentscheidungen bzw. Komponenten enthalten sind, ist sie anwendungsunabhängig. Allerdings ist die Anzahl der möglichen Anwendungsbereiche schon viel stärker eingeschränkt, als bei der

Technologieschicht, denn die Handhabungs- und Präsentationsschicht verkörpert eine gewisse Vorstellung von der Handhabung und der Präsentation, die sich an einem Leitbild mit Entwurfsmetaphern orientiert. Somit ist diese Schicht für die Konstruktion aller Anwendungen verwendbar, die das gleiche Leitbild verfolgen. Wenn allerdings ein anderes Leitbild den Rahmen für eine Anwendung vorgibt, so muß gegebenenfalls diese Schicht durch eine angemessenere ersetzt werden.



**Abbildung 48: Die WAM-Modellarchitektur ([GLL+99], S. 92)**

In der Abbildung 48 ist zu erkennen, daß die WAM-Modellarchitektur einen Rahmen für die zu entwickelnde fachliche Anwendung darstellt. Dabei stützt sich die Anwendung sowohl auf die Technologieschicht als auch auf die Handhabungs- und Präsentationsschicht ab, während sie keinen Zugriff auf die Systembasisschicht hat.

Um die Modellarchitektur anwendungsspezifischer zu gestalten, können auch an der Stelle der fachlichen Anwendung domänenspezifische Schichten vorgesehen werden (vgl. [BGK+97]). Dadurch kann eine optimale anwendungsfachliche Rahmenwerksunterstützung für den Entwickler erreicht werden.

Im folgenden möchte ich auf die eingangs erwähnten Probleme bei der Verwendung von Rahmenwerken zurückkommen und erläutern, wie mit der WAM-Modellarchitektur diese gelöst werden.

#### 1. *Unterstützung nur für eine fachlich beschränkte Domäne*

Dieses Problem wird zwar nicht direkt behoben, jedoch sehr entschärft. Auch ein Rahmenwerk mit der WAM-Modellarchitektur ist nur für eine fachlich beschränkte Domäne sinnvoll verwendbar. Allerdings ist es durch die Unterteilung in Schichten möglich, Teile des Rahmenwerkes in anderen Anwendungsbereichen wiederzuverwenden. In Abhängigkeit davon, wie sehr sich die verschiedenen Anwendungsbereiche unterscheiden, können ein, zwei oder alle Schichten wiederverwendet werden.

## 2. *Unübersichtlichkeit aufgrund zunehmender Komplexität*

Durch die logische Strukturierung in Schichten bleibt dem Anwendungsentwickler der Überblick erhalten. Darüber hinaus ist es ihm durch das geschaffene Verständnis über den groben Aufbau des Rahmenwerkes möglich, gezielt nach gewünschter Funktionalität zu suchen.

## 3. *Auswirkungen von Änderungen am Rahmenwerk auf das Anwendungssystem*

Wenn das Rahmenwerk in einer Schicht geändert wird, so kann aufgrund der definierten Abhängigkeiten zwischen den Schichten und dem Anwendungssystem genau festgestellt werden, welche Tragweite diese Änderung hat. Sollte z.B. die Systembasisschicht modifiziert werden, so wird aufgrund der WAM-Modellarchitektur schnell deutlich, daß sich diese Änderung höchstens auf die höheren Schichten auswirken kann, jedoch nicht auf das Anwendungssystem.

## 4. *Schwer auffindbare Entwurfsentscheidungen*

Gryczan et al. nennen als ein Ziel der WAM-Modellarchitektur:

"Diese logische Modellarchitektur soll dem Entwickler eine prinzipielle Orientierung über den Ort von Entwurfsentscheidungen geben." ([GLL+99], S. 93)

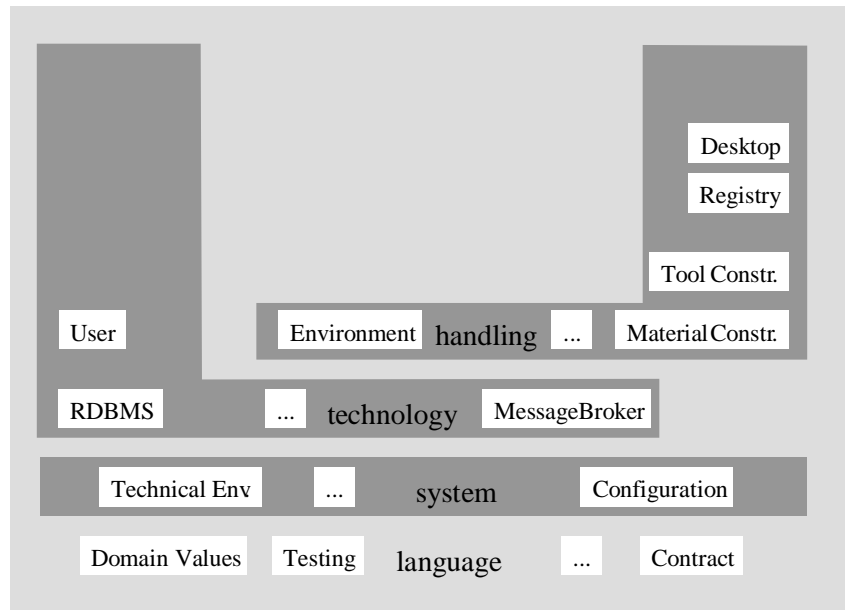
Dieses Ziel wird erreicht, indem durch die WAM-Modellarchitektur genau festgelegt wird, welche Art von Funktionalität in den Schichten untergebracht ist und wie sie verknüpft werden kann. Dadurch ist es für den Anwendungsentwickler einfach, Entwurfsentscheidungen zu lokalisieren und zu bestimmen, wo neue Funktionalität angesiedelt werden kann.

## **6.2 Architektur des JWAM-Rahmenwerkes und Einordnung des Postversandsystems**

Die Architektur des JWAM-Rahmenwerkes orientiert sich an der WAM-Modellarchitektur und besteht aus den drei Schichten<sup>17</sup> Systembasisschicht, Technologieschicht sowie Handhabungs- und Präsentationsschicht (siehe Abbildung 49). Neben diesen drei Schichten bietet das JWAM-Rahmenwerk noch eine Spracherweiterung für die Programmiersprache JAVA an, die von allen Schichten genutzt wird und auch für die zu entwickelnden Anwendungen zur Verfügung steht. Es enthält keine anwendungsspezifischen Komponenten, jedoch können anwendungsspezifische Schichten auf das JWAM-Rahmenwerk aufgesetzt werden, um es für die gegebenen Anforderungen angemessen zu spezialisieren. Ansonsten enthält das JWAM-Rahmenwerk alle notwendigen Komponenten, um interaktive Anwendungen zu bauen. In Abbildung 49 ist eine repräsentative Auswahl an Sub-Rahmenwerken aufgeführt und den Schichten zugeordnet, auf die ich im folgenden kurz eingehen möchte.

---

<sup>17</sup> Die Namen der Schichten sind im JWAM-Rahmenwerk in Englisch.



**Abbildung 49: Schichtenarchitektur des JWAM-Rahmenwerkes**

Die Spracherweiterung (language) enthält u.a. folgende wesentliche Sub-Rahmenwerke:

- *Domain Values*

Fachwerte sind ein zentrales Konzept des WAM-Ansatzes (vgl. [Zül98], S. 316ff.). Sie werden im gesamten JWAM-Rahmenwerk verwendet. Mit Hilfe dieser Klassen können eigene Fachwerte entwickelt werden. So sind im Postversandssystem die Transportbeschreibung (siehe Abschnitt 5.2.2) und die Transportadresse (siehe Abschnitt 5.2.1) als Fachwerte realisiert.

- *Testing*

Um ein automatisiertes Testen der entwickelten Klassen zu ermöglichen, können Testklassen geschrieben werden. Diese prüfen, ob eine Klasse ihrer Spezifikation genügt. Für alle wesentlichen Klassen des JWAM-Rahmenwerkes sowie des Postversandsystems gibt es korrespondierende Testklassen.

- *Contract*

Mit Hilfe des Vertragsmodells (vgl. [Mey97]) läßt sich die Softwarequalität sichern. In allen Klassen des JWAM-Rahmenwerkes wird ein Vertragsmodell eingesetzt. Weiterhin sind auch die Umgangsformen des Postversandsystems über Vor- und Nachbedingungen abgesichert. Dadurch läßt sich schnell eine Verletzung der Spezifikation feststellen.

Als unterste Schicht besteht die Systembasisschicht (system) u.a. aus den folgenden Sub-Rahmenwerken:

- *Technical Environment*

Eine leistungsfähige Ausnahmebehandlung sowie die Ausgabe von Systemmeldungen werden durch dieses Sub-Rahmenwerk unterstützt.

- *Configuration*

Beim Starten von Anwendungssystemen ist es oftmals notwendig, daß eine Reihe von Einstellungen gesetzt werden. So ist es z.B. auch beim Postversandssystem nötig, zu konfigurieren, auf welchem Computer im Netzwerk sich die Postdienststelle befindet.

Auf die Systembasisschicht setzt die Technologieschicht (technology) mit u.a. den folgenden Sub-Rahmenwerken auf:

- *User*

Eine Benutzerverwaltung ist in den meisten Anwendungssystemen unerlässlich, insbesondere wenn sie in kooperativen Arbeitsumgebungen eingebettet sind. Für das Postversandssystem wird die Benutzerverwaltung zur Identifikation des Benutzers am Arbeitsplatz benötigt.

- *MessageBroker*

Hinter dem MessageBroker verbirgt sich eine komfortable Nachrichtenvermittlung, die eine Interprozeßkommunikation erlaubt. Damit ist er sehr nützlich für die Realisierung verteilter Anwendungen. Auch im Postversandssystem wird der MessageBroker verwendet. Er dient in diesem Fall als Nachrichtenvermittlung zwischen dem Postzustelldienst- Server und -Client.

- *RDBMS*

Eine wichtige Anforderung in vielen Anwendungssystemen ist, daß Objekte persistent gespeichert werden können. Dieses Sub-Rahmenwerk unterstützt das persistente Speichern, indem es ein relationales Datenmodell anbietet, das von einer konkreten relationalen Datenbank abstrahiert.

Als oberste Schicht enthält die Handhabungs- und Präsentationsschicht (handling) u.a. die folgenden Sub-Rahmenwerke:

- *Environment*

Die Umgebung stellt den privaten Bereich dar und grenzt ihn gegenüber den Arbeitsumgebungen anderer Benutzer sowie den allgemein zugreifbaren Materialien ab. Sie enthält den Arbeitsplatz sowie alle Werkzeuge, Materialien und Automaten, die der Benutzer für seine Tätigkeit unmittelbar benötigt. Alle Werkzeuge des Postversandsystems sowie die Handhabungskomponente (siehe Abschnitt 5.3.1) und der Postzustelldienst-Client (siehe Abschnitt 5.3.2.1) befinden sich zur Laufzeit des Anwendungssystems in der Arbeitsumgebung des Benutzers.

- *Material Construction*

Materialien sind fachlich motivierte Gegenstände, die mit Hilfe von Werkzeugen bearbeitet werden. Die grundlegenden Entwurfsentscheidungen für die Entwicklung eigener Materialien sind in diesem Sub-Rahmenwerk enthalten. Beispielhafte Materialien des Postversandsystems sind die transportierbaren Materialien, die Postein- und Postausgangskörbe (siehe Abschnitt 5.3.1.4) sowie der Transportbehälter (siehe Abschnitt 5.3.1.5).



- *Tool Construction*

Dieses Rahmenwerk enthält die grundsätzlichen Mechanismen sowie Entwurfsmuster für den Bau von Werkzeugen. So basieren auch die Werkzeuge des Postversandsystems (siehe Abschnitt 5.4) auf diesem Sub-Rahmenwerk.

- *Registry*

Die Registratur ist eine fachlich motivierte Komponente zum persistenten Verwalten von Materialien in kooperativen Arbeitsumgebungen.

- *Desktop*

Der Desktop visualisiert grafisch den Arbeitsplatz des Benutzers (vgl. [Lip99]) und unterstützt den Umgang mit Materialien, Werkzeugen und Automaten. Der Desktop läßt sich mit den verschiedensten fachlich motivierten Gegenständen ausstatten, so auch mit den Postkörben und den Werkzeugen des Postversandsystems.

Es stellt sich nun noch die Frage, in welcher Schicht des JWAM-Rahmenwerkes das Postversandsystem angesiedelt werden kann. Diese Frage läßt sich mit der *Handhabungs- und Präsentationsschicht* beantworten. Wie bereits festgestellt wurde, enthält diese Schicht u.a. Komponenten, die den Umgang des Benutzers mit dem Anwendungssystem ausgestalten (siehe Abschnitt 6.1). Dabei orientieren sich die Komponenten an einem Leitbild mit Entwurfsmetaphern.

Das Postversandsystem ist eine Komponente, die der Ausstattung des Arbeitsplatzes dient. Daher prägt es auch den Umgang des Benutzers mit dem Anwendungssystem. Der Umgang orientiert sich am Leitbild vom Arbeitsplatz für eigenverantwortliche Expertentätigkeit, ebenso wie bei den Sub-Rahmenwerken der Handhabungs- und Präsentationsschicht des JWAM-Rahmenwerkes. Deswegen habe ich das Postversandsystem in diese Schicht integriert.



## 7 Zusammenfassung und Ausblick

Dieses Kapitel reflektiert die Argumentation dieser Arbeit und faßt die jeweiligen Ergebnisse zusammen. Weiterhin gehe ich darauf ein, wie die drei Arbeiten im Rahmen des COJAC-Projektes (siehe Abschnitt 1.1) verknüpft wurden und was an Themen offengeblieben ist.

### Argumentation und Ergebnisse

Das zentrale Thema dieser Arbeit war die Konzeption und Realisierung eines Softwaresystems zur Unterstützung kooperativer Arbeit mittels Materialaustausch und dessen Integration ins JWAM-Rahmenwerk. Dieses Thema wurde von mir im Rahmen des COJAC-Projektes vor dem Hintergrund der Softwareentwicklungsmethode WAM bearbeitet. Dabei war ein Schwerpunkt, daß das Softwaresystem als Basis für verschiedene aber ähnliche Kooperationsituationen einsetzbar sein soll. Um den Gegenstand dieses Softwaresystems zu verdeutlichen, habe ich es in Kapitel 1 Postversandsystem genannt.

Da der Gegenstand des Postversandsystems die Unterstützung kooperativer Arbeit ist, habe ich in Kapitel 2 eine Definition kooperativer Arbeit diskutiert. Dabei habe ich Möglichkeiten für eine evtl. Unterstützung hervorgehoben. Anschließend bin ich auf zwei Klassifikationen von Groupware eingegangen, die der Systematisierung und Abgrenzung kooperationsunterstützender Software dienen.

Die erste Klassifikation differenziert nach der Art der Unterstützung. Es wurden in diesem Zusammenhang die Kommunikations-, die Koordinations- und die Kooperationsunterstützung identifiziert. Das Postversandsystem konnte ich bei dieser Klassifikation sowohl der Kooperations- als auch der Kommunikationsunterstützung zuordnen. Weiterhin habe ich festgehalten, daß es der Koordinationsunterstützung als Grundlage dient.

Die zweite Klassifikation, die Time-Space-Matrix, differenziert nach den örtlichen und zeitlichen Rahmenbedingungen. Das Postversandsystem ließ sich in dieser Matrix in dem Bereich der asynchronen verteilten Interaktion einordnen. Darunter sollen solche Kooperationsituationen verstanden werden, bei denen sich die Kooperationspartner zu unterschiedlichen Zeiten an verschiedenen Orten befinden.

Durch die Klassifizierung des Postversandsystems konnte ich eingrenzen, welche Kooperationsituationen es unterstützen soll. Dabei wurde deutlich, daß es nahezu das selbe Spektrum einer herkömmlichen elektronischen Post abdeckt. Mit diesem Ausgangspunkt habe ich in Kapitel 4 die fachlichen und die technischen Anforderungen an das Postversandsystem aufgestellt und die Unterschiede zur elektronischen Post diskutiert. In Kapitel 5 bin ich darauf eingegangen, wie der Entwurf diese Anforderungen erfüllt.

Folgende fachliche Anforderungen sowie Ergebnisse konnte ich festhalten, in denen sich das Postversandsystem von einer herkömmlichen elektronischen Post unterscheiden soll:

#### 1. *Asynchrone verteilte Kooperation mit Hilfe von Kooperationsmitteln und -medien*

Das Postversandsystem soll Kooperationsituationen unterstützen, bei der sich die Benutzer eines Anwendungssystems zu unterschiedlichen Zeiten an verschiedenen Orten befinden können. Dazu soll es Kooperationsmittel und -medien anbieten, mit denen die Arbeitsplätze der Benutzer ausgestattet werden können.

Um die asynchrone verteilte Kooperation zu unterstützen, ermöglicht das Postversandsystem den Austausch von Materialien zwischen verschiedenen Arbeitsplätzen und erlaubt dem Empfänger, zu einem beliebigen Zeitpunkt auf die empfangenen Materialien zuzugreifen.

Das Postversandsystem stellt Postkörbe zur Verfügung, um es als Kooperationsmedium auf dem Arbeitsplatz des Benutzers zu vergegenständlichen. Als Kooperationsmittel werden Transportbehälter angeboten.

## 2. *Verschiedenste Materialtypen können zwischen den Arbeitsplätzen ausgetauscht werden*

Während mit einer herkömmlichen elektronischen Post nur Textnachrichten mit Dateianhang versendet werden können, soll das Postversandsystem verschiedenste Typen von Materialien erlauben. Dadurch wird ein anwendungsfachlich motivierter Umgang mit den Materialien ermöglicht, insbesondere beim Versenden und Empfangen. Außerdem werden somit Fehlerquellen aufgrund der Datenkonvertierung in ein allgemeineres Format vermieden.

Diese Anforderung wird erfüllt, indem das Postversandsystem nicht vorgibt, was versendet werden kann. Es setzt lediglich voraus, daß die zu transportierenden Materialien eine vorgegebene Schnittstelle erfüllen.

## 3. *Einheit von Ort und Zeit der Materialien*

Das Postversandsystem soll sicherstellen, daß sich ein Material zu einem Zeitpunkt nur an einem Ort befinden kann.

Um das zu realisieren, habe ich vorgesehen, daß ein Material immer nur an einen Empfänger zeitgleich gesendet werden kann. Ansonsten wird die Einheit von Ort und Zeit der Materialien sichergestellt, indem das Postversandsystem ein abgesendetes Material vom Arbeitsplatz des Senders entfernt und in der Postdienststelle zwischenspeichert. Wenn zu einem späteren Zeitpunkt der Empfänger das Material anfordert, dann wird es aus der Postdienststelle entnommen und zu dem Arbeitsplatz des Empfängers übertragen.

## 4. *Unterstützung verschiedener Benutzungsmodelle*

Die Zielsetzung besagt, daß das Postversandsystem als Basis in verschiedenen aber ähnlichen Kooperationssituationen einsetzbar sein soll. Als Anforderung folgt daraus, daß sich die Benutzungsmodelle an die jeweilige Situation adaptieren lassen müssen.

Um diese Anforderung zu erfüllen, habe ich dafür speziell eine Handhabungskomponente vorgesehen. Die von mir entwickelte Komponente kann für jeden Benutzer individuell konfiguriert werden. Sollten jedoch die Konfigurationsmöglichkeiten nicht ausreichend sein, so kann diese Komponente durch eine anwendungsspezifischere ersetzt werden, um den Anwendungskontext angemessen zu unterstützen.

Eine weitere Anforderung bestand darin, eine Vorstellung von der Handhabung und Präsentation des Postversandsystems zu schaffen. Dieser Anforderung bin ich gerecht geworden, indem ich die drei folgenden verschiedenen Benutzungsmodelle dargestellt habe, die das Postversandsystem aufweisen soll. Das erste Benutzungsmodell verkörpert einen sehr einfachen Umgang, der für die meisten Anwendungsfälle wahrscheinlich ausreicht, aber keine individuellen Anpassungen erlaubt. Aus diesem Grund habe ich einen erweiterten Umgang beschrieben, der an die jeweiligen Bedürfnisse der Benutzer angepaßt werden kann. Um diese

Anpassungen durchzuführen, bedarf es einer Konfiguration des Postein- und Postausgangs. Wie diese Konfiguration erfolgen kann, habe ich im dritten Benutzungsmodell beschrieben.

Weiterhin habe ich technische Anforderungen aufgezeigt, die sich auf die Softwarearchitektur des Postversandsystems beziehen. Eine dieser technischen Anforderungen war, daß das Postversandssystem als Groupware nutzbar sein soll. Dieser Anforderung wurde entsprochen, weil die Softwarearchitektur eine black-box-artige Verwendung erlaubt, indem es für alle variablen Aspekte vorgefertigte Standardlösungen anbietet. Somit kann die Softwarearchitektur unmittelbar als Mehrbenutzer-Software zur Unterstützung kooperativer Arbeit genutzt werden.

Eine weitere technische Anforderung war, daß die Softwarearchitektur Eigenschaften aufweisen soll, damit das Postversandssystem als Basisunterstützung wiederverwendet werden kann. Als Ergebnis konnte ich festhalten, daß die Softwarearchitektur ein Rahmenwerk zur Unterstützung expliziter asynchroner verteilter Kooperation mittels Materialaustausch sein soll, daß über die Eigenschaften Erweiterbarkeit, Anpaßbarkeit, Wiederverwendbarkeit und Kombinierbarkeit verfügt. Um all diese Eigenschaften zu realisieren, habe ich die verwendeten Schnittstellen minimal gehalten, Hot-Spots für anwendungsspezifische Anpassungen vorgesehen und die Softwarearchitektur in sinnvoll abgrenzbare Komponenten unterteilt.

In Kapitel 5 habe ich den Entwurf des Postversandsystems vorgestellt, der sich an der Entwurfsmetapher Hauspost orientiert. Er berücksichtigt die virtuelle Natur der im Anwendungssystem modellierten Arbeitsplätze, indem die Materialien über eine zentrale Stelle den Empfängern zugestellt werden. Somit habe ich im Entwurf eine Aufteilung des Postversandsystems in die folgenden drei Komponenten vorgesehen, die verschiedene Aufgaben wahrnehmen:

1. *Postdienststelle*

Sie ist die zentrale Stelle, welche die zu transportierenden Materialien zwischenspeichert und den Empfänger über den Erhalt von Materialien benachrichtigt.

2. *Postzustelldienst*

Diese Komponente kapselt die Interprozeßkommunikation und ist für den Transport der Materialien sowie Benachrichtigungen zwischen den Arbeitsplätzen der Benutzer und der Postdienststelle zuständig.

3. *Handhabungskomponente*

Die Handhabungskomponente verkörpert die Benutzungsmodelle des Postversandsystems. Sie ist damit die Komponente, die bestimmt, wie anwendungsfachlich motiviert das Versenden und Empfangen von Materialien vonstatten gehen.

Diese drei Komponenten ließen sich in den in Kapitel 2 vorgestellten Klassifizierungen einordnen. So ist die Funktionalität aller Komponenten zusammen so ausgelegt, daß sie die asynchrone verteilte Interaktion unterstützen. Weiterhin ließen sich die Komponenten Postzustelldienst zusammen mit der Postdienststelle der Kommunikationsunterstützung zuordnen, während die Handhabungskomponente darauf aufbauend der Kooperationsunterstützung dient.

In Kapitel 6 bin ich schließlich darauf eingegangen, wie das Postversandsystem in das JWAM-Rahmenwerk integriert wurde. Dazu habe ich zuerst die WAM-Modellarchitektur vorgestellt. Sie schlägt die Strukturierung von Rahmenwerken in die Systembasis-, die Technologie- sowie die Handhabungs- und Präsentationsschicht vor. Diese Modellarchitektur war maßgeblich für die Entwicklung des JWAM-Rahmenwerkes, was bei dessen Beschreibung deutlich wurde. Als Ergebnis konnte ich festhalten, daß aufgrund der WAM-Modellarchitektur und des daraus resultierenden Aufbaus des JWAM-Rahmenwerkes ich das Postversandsystem in der Handhabungs- und Präsentationsschicht integriert habe.

### **Vernetzung der Arbeiten des COJAC-Projektes**

In Kapitel 1 hatte ich beschrieben, daß im Rahmen des COJAC-Projektes drei Arbeiten entstanden sind. Die vorliegende Arbeit ist eine von denen. Die anderen Arbeiten waren die Entwicklung eines Raumsystems und die Unterstützung komplexer Kooperation mit Hilfe von Vorgangsmappen sowie einer Vorgangsverfolgung. Im folgenden möchte ich darauf eingehen, wie diese Arbeiten miteinander vernetzt werden können bzw. wurden.

#### *Vernetzung der Vorgangsmappen und der Vorgangsverfolgung mit dem Postversandsystem*

Die Realisierung von Vorgangsmappen und Laufzetteln steht in einem unmittelbaren Zusammenhang mit dem Postversandsystem, da Vorgangsmappen zwischen verschiedenen Arbeitsplätzen ausgetauscht werden. Um den Austausch zu gewährleisten, wird das Postversandsystem verwendet. Damit das Postversandsystem eine Vorgangsmappe mit dem Laufzettel transportieren kann, erfüllt sie die Schnittstelle eines transportierbaren Materials (siehe Abschnitt 5.2.2). Hinter dieser Schnittstelle wird aus der Sicht des Postversandsystems der Laufzettel verborgen, der die Empfängeradresse der Vorgangsmappe enthält.

Auch die Vorgangsverfolgung ist eng an das Postversandsystem gekoppelt, da sie u.a. protokolliert, wann und wo welche Vorgangsmappe abgesendet und empfangen wurde. Eine Möglichkeit zur Kopplung der Vorgangsverfolgung an das Postversandsystem besteht darin, spezielle Postein- und Postausgangsautomaten (siehe Abschnitt 5.3.1) zu schaffen. Diese versorgen die Vorgangsverfolgung mit entsprechenden Informationen, wenn sie eine Vorgangsmappe über den Postzustelldienst versenden bzw. erhalten.

#### *Vernetzung des Raumsystems mit dem Postversandsystem*

Die Vernetzung dieser beiden Systeme ist nicht zwingend, da sie unabhängig voneinander sind. Dennoch ergänzen sie sich in verschiedenen Hinsichten hervorragend. So ist es bei dem Postversandsystem bis jetzt so, daß die Postdienststelle als Dienstleistungsanbieter fachlich unmotiviert auf irgendeinem zentral zugreifbaren Computer im Gesamtsystem angesiedelt ist.

Besser wäre jedoch, wenn für die Postdienststelle ein eigener Raum im Raumsystem vorgesehen wird. Dadurch wäre der Ort der Postdienststelle im Anwendungssystem fachlich motiviert. So könnte man vorsehen, daß nur ein Administrator Zugang zu diesem Raum hat, um direkt bei der Postdienststelle neue Adressen anzumelden bzw. alte zu löschen.

Eine weitere Möglichkeit der Vernetzung besteht in der Ausgestaltung von Räumen mit Postkörben. Insbesondere bei Privaträumen erscheint diese Möglichkeit vorteilhaft, da somit fachlich motiviert Materialien zwischen verschiedenen Privaträumen ausgetauscht werden könnten, auf die sonst kein Zugriff bestünde.

Aber auch die Ausgestaltung von öffentlichen Räumen erscheint sinnvoll. So könnte sich z.B. in einem Gruppenraum ein Posteingangskorb befinden. Jeder der Gruppenmitglieder hätte

Zugriff auf diesen Raum und somit auch auf den Posteingangskorb. Wenn nun ein Material vom Postzustelldienst in diesem Posteingangskorb abgelegt wird, so könnte sich ein Gruppenmitglied dieses Material entnehmen. Über diese Ausgestaltungsvariante besteht die Möglichkeit, explizite Kooperation mittels Materialaustausch zwischen Gruppen bzw. Gruppenarbeitsplätzen zu unterstützen.

Leider konnte die Vernetzung des Raumsystems und des Postversandsystems nicht realisiert werden. Der Grund dafür ist, daß sich das JWAM-Rahmenwerk in einer stetigen Weiterentwicklung befindet, wobei die einzelnen Versionen inkompatibel zueinander sind. So wurde das Postversandsystem immer an die aktuelle Version des JWAM-Rahmenwerkes angepaßt, während das Raumsystem in diesem Entwicklungsprozeß nicht mitgepflegt wurde. Folglich funktionierten das Raumsystem und das Postversandsystem nur mit verschiedenen Versionen des JWAM-Rahmenwerkes und konnten zum Zeitpunkt der vorliegenden Arbeit miteinander nicht in Einklang gebracht werden.

### **Ausblick**

Wie ich oben bereits erwähnte, konnte das Postversandsystem aus technischen Gründen nicht mit dem Raumsystem vernetzt werden. Aber gerade in dieser Vernetzung sehe ich ein großes Potential weiterführender Kooperationsunterstützungsmöglichkeiten.

Neben der Betrachtung zusätzlicher Kooperationssituationen, bietet das Postversandsystem weitere Entwicklungsmöglichkeiten. So wird bisher die Adressierung funktioneller Rollen nicht unterstützt, da das JWAM-Rahmenwerk nur eine sehr rudimentäre Benutzerverwaltung anbietet, die keine Rollen vorsieht. Gegenwärtig wird aber gerade an einer umfassenden Benutzerverwaltung gearbeitet, die auch ein Rollenkonzept zur Verfügung stellen soll. Somit könnte darauf aufbauend das Postversandsystem in seiner Funktionalität so erweitert werden, daß auch die explizite Kooperation zwischen funktionellen Rollen unterstützt wird.

Das Postversandsystem erlaubt bis jetzt keine Mehrfachadressierung von Materialien. Eine Weiterentwicklung könnte folglich darin bestehen, eine Mehrfachadressierung zu unterstützen. Damit einher geht, daß auch ein Konzept von Original und Kopie vorgesehen werden müßte, damit die Einheit von Ort und Zeit der Materialien weiterhin gewährleistet bleibt. Dieses könnte so umgesetzt werden, daß ein bestimmter Empfänger das Original erhält, während für all die anderen Kopien erzeugt und zugestellt werden.





## Abbildungsverzeichnis

Abbildung 1: Auswahl verschiedener Kooperationsmittel und -medien des COJAC-Projektes .....	2
Abbildung 2: Grafische Hervorhebung von Ein- und Ausgangsschnittstellen einer Komponente .....	5
Abbildung 3: Struktur der Unterstützung ([Bur97], S. 39) .....	9
Abbildung 4: Time-Space-Matrix ([EGR91], S. 41).....	17
Abbildung 5: Würfel der Unterstützung.....	20
Abbildung 6: Einordnung des Postversandsystems.....	21
Abbildung 7: Leitbilder in der Softwareentwicklung ([Zül98], S. 74) .....	26
Abbildung 8: Anforderungen an das Postversandsystem im Vergleich zu einer herkömmlichen elektronischer Post .....	34
Abbildung 9: Einfaches Benutzungsmodell.....	36
Abbildung 10: Erweitertes Benutzungsmodell.....	37
Abbildung 11: Benutzungsmodell der Konfiguration des Posteingangs.....	39
Abbildung 12: Typische Struktur eines Hot-Spots.....	43
Abbildung 13: Konzeptioneller Entwurf des Postversandsystems.....	50
Abbildung 14: Umgangsformen der Klasse Transportadresse.....	54
Abbildung 15: Einordnung eines transportierbaren Materials aus der Sicht des Postversandsystems .....	56
Abbildung 16: Umgangsformen des Aspektes TransportierbaresMaterial .....	58
Abbildung 17: Umgangsformen der Klasse Transportbeschreibung.....	60
Abbildung 18: Umgangsformen des Aspektes AdressierbaresMaterial .....	61
Abbildung 19: Struktur eines einfachen Materials.....	62
Abbildung 20: Darstellung von Komponenten mit ihren Ein- und Ausgangsschnittstellen ...	64
Abbildung 21: Verwendung der Schnittstelle Benachrichtbar.....	65
Abbildung 22: Umgangsform der Schnittstelle Benachrichtbar .....	66
Abbildung 23: Ein- und Ausgangsschnittstellen der Handhabungskomponente .....	67
Abbildung 24: Aufbau der Handhabungskomponente.....	68
Abbildung 25: Aufbau des Posteingangsautomaten.....	69
Abbildung 26: Aufbau der Posteingangsstrategien .....	70
Abbildung 27: Umgangsformen des Postausgangsautomaten .....	72
Abbildung 28: Icons der Postkörbe (von links nach rechts: leerer und befüllter Postausgangskorb, leerer und befüllter Posteingangskorb).....	72
Abbildung 29: Aufbau des Postein- und Postausgangskorbes .....	73
Abbildung 30: Aufbau des Transportbehälters .....	74
Abbildung 31: Aufbau und Funktionsweise der Rohrpost.....	74
Abbildung 32: Ein- und Ausgangsschnittstellen des Postzustelldienstes .....	76
Abbildung 33: Umgangsformen des Postzustelldienstes .....	77
Abbildung 34: Aufbau des Postzustelldienstes .....	78
Abbildung 35: Aufbau und Umgangsformen des Postzustelldienst-Client .....	79
Abbildung 36: Umgangsformen des Postzustelldienst-Servers .....	80
Abbildung 37: Ein- und Ausgangsschnittstellen der Postdienststelle .....	80
Abbildung 38: Umgangsformen der Postdienststelle.....	81
Abbildung 39: Aufbau der Postdienststelle.....	82
Abbildung 40: Aufbau und Umgangsformen der Postfachwand mit Postfächern .....	83
Abbildung 41: Der Arbeitsplatz .....	86

Abbildung 42: Der Adreßeditor .....	87
Abbildung 43: Der Posteingangskorbeditor .....	88
Abbildung 44: Der Postausgangskorbeditor.....	89
Abbildung 45: Das Einstellwerkzeug des Posteingangsautomaten.....	89
Abbildung 46: Das Einstellwerkzeug der Standardstrategie.....	90
Abbildung 47: Das Einstellwerkzeug der Reservierungsstrategie .....	91
Abbildung 48: Die WAM-Modellarchitektur ([GLL+99], S. 92).....	97
Abbildung 49: Schichtenarchitektur des JWAM-Rahmenwerkes.....	99

## Literaturliste

[Bäu98]

D. Bäumer: Softwarearchitekturen für die rahmenwerkbasierte Konstruktion großer Anwendungssysteme. Dissertationsschrift am Fachbereich Informatik der Universität Hamburg, 1998.

[BBK+99]

W. Becker, C. Burger, J. Klarmann, O. Kulendik, F. Schiele, K. Schneider: Rechnerunterstützung für Interaktionen zwischen Menschen. In: Informatik Spektrum, Springer, Heidelberg, Band 22, Heft 6, S. 422-435, Dezember 1999.

[BGK+97]

D. Bäumer, G. Gryczan, R. Knoll, C. Lilienthal, D. Riehle, H. Züllighoven: Framework Development for Large System. Communications of the ACM, Vol. 40 No. 10, pp. 52-59, 1997.

[Bre00]

H. Breitling: Unterstützung kooperativer Arbeitsprozesse: Vorgangsmappen im JWAM-Framework. Diplomarbeit am Fachbereich Informatik der Universität Hamburg, 2000.

[BRJ99]

G. Booch, J. Rumbaugh, I. Jacobson: Das UML-Benutzerhandbuch. Addison-Wesley, Bonn, 1999.

[Bro90]

Brockhaus-Enzyklopädie. 19. Auflage, Brockhaus, Mannheim, Band 12, 1990.

[Bur97]

C. Burger: Groupware - Kooperationsunterstützung für verteilte Anwendungen. dpunkt.-Verlag, Heidelberg, 1997.

[CHS+97]

W. Codenie, K. De Hondt, P. Steyaert, A. Vercammen: From Custom Applications to Domain-Specific Frameworks. Communications of the ACM, Vol. 40 No. 10, pp. 71-78, 1997.

[CSCW86]

Proceedings of ACM CSCW'86 Conference on Computer Supported Cooperative Work, Austin, Texas, December 1986. ACM, New York.

[CSCW88]

Proceedings of ACM CSCW'88 Conference on Computer Supported Cooperative Work, Portland, Oregon, September 1988. ACM, New York.

[CSCW90]

Proceedings of ACM CSCW'90 Conference on Computer Supported Cooperative Work, Los Angeles, California, October 1990. ACM, New York.

[DMN+97]

S. Demeyer, T. D. Meijler, O. Nierstrasz, P. Steyaert: Design Guidelines for 'Tailorable' Frameworks. Communications of the ACM, Vol. 40, No. 10, pp. 60-64, 1997.

[Dud93]

Duden "Informatik" - Ein Sachlexikon für Studium und Praxis. DUDENVERLAG, Mannheim, 1993.

[ECSCW89]

Proceedings of the First European Conference on Computer-Supportes Cooperative Work, London, UK, September 1989.

[ECSCW91]

L. Bannon, M. Robinson & K. Schmidt (Hrsg.). Proceedings of the Second European Conference on Computer-Supported Cooperative Work, Amsterdam, Niederlande, September 1991.

[ECSCW93]

G. De Michelis, C. Simone & K. Schmidt (Hrsg.). Proceedings of the Third European Conference on Computer-Supported Cooperative Work, Milan, Italien, September 1993.

[EGR91]

C. A. Ellis, S. J. Gibbs, G. L. Rein: Groupware - Some Issues and Experiences. Communications of the ACM, Vol. 34 No. 1, pp. 38-58, 1991.

[Flo84]

C. Floyd: A Systematic Look at Prototyping. In: R. Budde, K. Kuhlenkamp, L. Mathiassen und H. Züllighoven (Hrsg.). Approaches to Prototyping. Springer, Berlin, pp. 1-18, 1984.

[Fre98]

M. Freund: Unterstützung kooperativer Arbeit mit Hilfe eines Transportsystems. Studienarbeit am Fachbereich Informatik der Universität Hamburg, 1999.

[GHJ+95]

E. Gamma, R. Helm, R. Johnson, J. Vlissides: Design Patterns - Elements of Reusable Object-Oriented Software. Addison-Wesley, Reading, Massachusetts, 1995.

[GHR+00]

G. Gryczan, A. Havenstein, S. Roock, I. Wetzel, H. Züllighoven: Kooperation mit persistenten fachlichen Behältern. In: OBJEKTspektrum, SIGS Conferences GmbH, Heft 1, S. 82-87, Januar/Februar 2000.

[GKZ94]

G. Gryczan, K. Kilberth, H. Züllighoven: Objektorientierte Anwendungsentwicklung. 2. Auflage, Vieweg, Braunschweig, Wiesbaden, 1994.

[GLL+99]

G. Gryczan, C. Lilienthal, M. Lippert, S. Roock, H. Wolf, H. Züllighoven: Framework-basierte Anwendungsentwicklung (Teil 1). In: OBJEKTSpektrum, SIGS Conferences GmbH, Heft 1, S. 90-98, Januar/Februar 1999.

[Gru94]

J. Grudin: CSCW: History and Focus. In: Computer, Vol. 27, No. 5, IEEE Computer Society, Los Alamitos, pp. 19-26, May 1994.

[Gry96]

G. Gryczan: Prozeßmuster zur Unterstützung kooperativer Tätigkeit. Deutscher Universitätsverlag, Wiesbaden, 1996.

[Hav99]

A. Havenstein: Unterstützung Kooperativer Arbeit durch eine Softwareregistratur. Studienarbeit am Fachbereich Informatik der Universität Hamburg, 1999.

[JBS97]

S. Jablonski, M. Böhm, W. Schulze (Hrsg.): Workflow-Management - Entwicklung von Anwendungen und Systemen. dpunkt.-Verlag, Heidelberg, 1997

[JF88]

R. E. Johnson, B. Foote: Designing Reuseable Classes. The Journal of Object-Oriented Programming, Vol. 1, No. 2, SIGS Publications, New York, pp. 22-35, June/July 1988.

[Ker93]

H. Kerner: Rechnernetze nach OSI. 2. Auflage, Addison-Wesley, Bonn, 1993.

[Koc00]

J. Koch: JWAM-Komponenten zur Kooperationsunterstützung: Die Raumkomponente. Diplomarbeit am Fachbereich Informatik der Universität Hamburg, 2000.

[Lip99]

M. Lippert: Die Desktop-Metapher in Systemen nach dem Werkzeug- und Material-Ansatz. Diplomarbeit am Fachbereich Informatik der Universität Hamburg, 1999.

[MaCr90]

T. Malone, K. Crowstone: What is Coordination Theory and How Can It Help Design Cooperative Work Systems? In: Proceedings of ACM CSCW'90 Conference on Computer-Supported Cooperative Work, Los Angeles, October 1990. ACM, New York, pp. 357-370, 1990.

[MC94]

T. Malone, K. Crowston: The Interdisciplinary Study of Coordination. ACM Computing Surveys, 26 (1994) 1, pp. 87-120.

[Mey97]

B. Meyer: Object-Oriented Software Construction (2<sup>nd</sup> ed.). Prentice Hall PTR, New Jersey, 1997.

[MO92]

S. Maaß, H. Oberquelle: Perspectives and Metaphors for Human-Computer Interaction. In: C. Floyd, H. Züllighoven, R. Budde, R. Keil-Slawik (Hrsg.). Software Development and Reality Construction. Springer, Berlin, pp. 233-251, 1992.

[Obe91]

H. Oberquelle: Kooperative Arbeit und menschengerechte Groupware als Herausforderung für die Software-Ergonomie. In: H. Oberquelle (Hrsg.). Kooperative Arbeit und Computerunterstützung - Stand und Perspektiven. Verlag für Angewandte Psychologie, Göttingen, 1991.

[Obe99]

H. Oberquelle: Computergestützte kooperative Arbeit. Vorlesungsunterlagen Wintersemester 1998/99, Universität Hamburg, Hamburg, 1999.

[PoBl96]

G. Pomberger, G. Blaschek: Software-Engineering: Prototyping und objektorientierte Software-Entwicklung. 2. Auflage, Carl Hanser, München, 1996.

[Rob93]

M. Robinson: Design for unanticipated use ..... In: G. De Michelis, C. Simone & K. Schmidt (Hrsg.). Proceedings of the Third European Conference on Computer-Supported Cooperative Work, Milan, Italien, September 1993.

[RoBl91]

T. Rodden, G. Blair: CSCW and Distributed Systems: The Problem of Control. In: L. Bannon, M. Robinson & K. Schmidt (Hrsg.). Proceedings of the Second European Conference on Computer-Supported Cooperative Work, Amsterdam, Niederlande, September 1991.

[RoWo98]

S. Roock, H. Wolf: Die Raummetapher zur Entwicklung kooperationsunterstützender Softwaresysteme für Organisationen. Diplomarbeit am Fachbereich Informatik der Universität Hamburg, 1998.

[Sam97]

J. Sametinger: Software Engineering with reusable components. Springer, Berlin, 1997.

[Sch97]

H. A. Schmidt: Systematic Framework Design by Generalization. Communications of the ACM, Vol. 40 No. 10, pp. 48-51, 1997.

[Sør88]

P. Sørgaard: Object Oriented Programming and Computerised Shared Material. In: K. Gjessing, C. Nygaard (Hrsg.). ECOOP'88, Springer, pp. 319-334, 1988.

[Syri97]

A. Syri: Tailoring Cooperation Support through Mediators. In: J. Hughes, W. Prinz, T. Rodden, K. Schmidt (Hrsg.). Proceedings of the Fifth European Conference on Computer Supported Cooperative Work, Niederlande, pp. 157-172, 1997.

[Szy98]

C. Szyperski: Component Software - Beyond Object-Oriented Programming. Addison-Wesley, Harlow, England, 1998.

[Wul95]

M. Wulf: Konzeption und Realisierung einer Umgebung zur Koordination rechnergestützter Tätigkeiten in kooperativen Arbeitsprozessen. Diplomarbeit am Fachbereich Informatik der Universität Hamburg, 1995.

[Zül98]

H. Züllighoven: Das objektorientierte Konstruktionshandbuch nach dem Werkzeug & Material-Ansatz. dpunkt.-Verlag, Heidelberg, 1998.