

Ein Ansatz zur Unterstützung wissensintensiver Prozesse durch Workflow-Management-Systeme

vorgelegt von
Diplom-Informatiker
Thomas Goesmann
aus Bochum

Von der Fakultät IV - Elektrotechnik und Informatik
der Technischen Universität Berlin
zur Erlangung des akademischen Grades
Doktor der Ingenieurwissenschaften
- Dr.-Ing. -

genehmigte Dissertation

Promotionsausschuss:

Vorsitzender: Prof. Dr. Bernd Mahr
Berichter: Prof. Dr. Herbert Weber
Berichter: Prof. Dr.-Ing. Stefan Jablonski

Tag der wissenschaftlichen Aussprache: 12.03.2002

Berlin 2002

D 83

Danksagung

Zunächst möchte ich Herrn Prof. Dr. Herbert Weber für die Schaffung des kreativen Umfelds am Fraunhofer-Institut für Software- und Systemtechnik ISST danken, in dem diese Arbeit entstehen konnte. Darüber hinaus danke ich ihm für seine Bereitschaft zur Diskussion und seine hilfreichen Anmerkungen und Beiträge.

Herrn Prof. Dr. Stefan Jablonski danke ich für die kritische Würdigung der Arbeit und seine kontinuierliche Diskussionsbereitschaft.

Besonderer Dank gilt Dr. Wolfgang Deiters, der durch seine Unterstützung und Motivation den gesamten Prozess der Erstellung dieser Arbeit von den ersten Ideen an begleitet hat und maßgeblichen Anteil am erfolgreichen Abschluss der Arbeit besitzt.

In diesem Zusammenhang möchte ich ebenfalls Dr.-Ing. Rüdiger Striemer danken, der mir insbesondere in den ersten Phasen der Arbeit immer wieder als Diskussionspartner zur Verfügung stand und wesentliche Beiträge zur Fokussierung des Themas lieferte.

Marcel Hoffmann vom Fachgebiet Informatik und Gesellschaft an der Universität Dortmund danke ich für seine Anmerkungen und Kommentare sowie für viele gute Ideen.

Weiterhin möchte ich mich bei allen Kollegen der Abteilung Prozessmanagement am Fraunhofer ISST für Diskussionen und Hilfestellungen bedanken, die zum Gelingen dieser Arbeit beigetragen haben.

Schließlich danke ich Mariele Hagen für die Korrektur der Arbeit und für vieles mehr.

Dortmund, im Oktober 2001

Thomas Goesmann

Inhalt

A	Einleitung	1
1	Einleitung und Aufgabenstellung	3
B	Grundlagen für eine Unterstützung wissensintensiver Prozesse durch Workflow-Management-Systeme	7
2	Grundlagen des Workflow Managements und des Wissensmanagements.....	9
2.1	Workflow-Management-Systeme.....	9
2.1.1	Überblick und Begriffsbestimmung	9
2.1.2	Der Prozesslebenszyklus	13
2.1.3	Verwaltung von prozessbezogenen Informationen in heutigen WFMS.....	15
2.1.4	Einsatzspektrum heutiger Workflow-Management-Systeme	20
2.2	Wissensmanagement.....	22
2.2.1	Überblick und Begriffsbestimmung	23
2.2.2	Bausteine des Wissensmanagements.....	26
2.2.3	Organizational Memory Information Systems (OMIS).....	29
2.2.4	Wissensarbeit.....	33
2.3	Dokumentenmodellierung	35
3	Ansätze zur Integration von Workflow Management und Wissensmanagement	39
3.1	Ein Überblick über prozessbezogenes Wissensmanagement.....	39
3.2	WorkBrain	43
3.3	Workware	46
3.4	Bramble.....	47
3.5	KnowMore	49
3.6	Ontobroker.....	51
3.7	EULE2.....	53
3.8	Milos	55

3.9	PROMOTE	56
3.10	Zusammenfassung und Bewertung der Ansätze.....	57
4	Wissensintensive Prozesse	61
4.1	Begriffsbildung	61
4.1.1	Einordnung und grundlegende Charakterisierung	61
4.1.2	Wissenstransfer zwischen Geschäftsfällen unterschiedlicher Prozesse.....	64
4.1.3	Bewahrung prozessbezogenen Wissens	67
4.1.4	Zusammenfassung	70
4.2	Beispiel für einen wissensintensiven Prozess: „Contract Management“	70
C	Konzeption einer Erweiterung von Workflow-Management-Systemen zur Unterstützung wissensintensiver Prozesse	75
5	Anforderungen an Workflow-Management-Systeme zur Unterstützung wissensintensiver Prozesse	77
5.1	Verwaltung prozessbezogenen Wissens	77
5.2	Unterstützung unterschiedlicher Funktionen und Charakteristik prozessbezogenen Wissens.....	79
5.3	Strukturierung sekundären Wissens anhand des Bearbeitungskontextes	81
5.4	Kooperativer Aufbau und Bewahrung sekundären Wissens.....	84
5.5	Integration der Verwaltung sekundären Wissens in den Prozesslebenszyklus	86
5.6	Teilaufgaben bei der Entwicklung eines Konzepts zur Unterstützung wissensintensiver Prozesse durch Workflow-Management-Systeme.....	87
6	Kontexte zur Unterstützung wissensintensiver Prozesse	89
6.1	Überblick	89
6.2	Identifikation von kontextbildenden Objekttypen	92
6.2.1	Untersuchung existierender Workflow-Metamodelle	92
6.2.2	Identifikation der kontextbildenden Objekttypen und Struktur der zugehörigen Kontexte.....	96

6.3	Ein Informationsmodell zur Beschreibung von Kontextdokumenten ..	105
6.3.1	Identifikation	105
6.3.2	Inhalt	107
6.3.3	Lebenszyklus	110
6.3.4	Adressaten und Berechtigungen	112
6.3.5	Sichtbarkeit	114
6.3.6	Bereitstellung	120
6.4	Eine Standardbibliothek für Informationen aus Workflow-Management-Systemen	126
7	Methodik zur Unterstützung wissensintensiver Prozesse durch Workflow-Management-Systeme	131
7.1	Integration von Maßnahmen zur Verwaltung von Kontexten in den Prozesslebenszyklus	131
7.1.1	Initialisierung von Kontexten	133
7.1.2	Nutzung von Kontexten	136
7.1.3	Evaluierung von Kontexten	140
7.2	Unterstützung des wissensintensiven Prozesses "Contract Management" durch Kontexte	141
D	Informationstechnische Umsetzung und Realisierung eines erweiterten Workflow-Management-Systems zur Unterstützung wissensintensiver Prozesse	145
8	Erweiterung von Workflow-Management-Systemen zur Unterstützung von Kontexten	147
8.1	Anforderungsanalyse	147
8.1.1	Anforderungen an WFMS	147
8.1.2	Anwendungsfallanalyse	148
8.2	Systemarchitektur	150
8.2.1	Übersicht	150
8.2.2	Kontext-Client	152
8.2.3	Workflow-Client	156
8.2.4	Kontext-Server	157

	8.2.5 Kontextdatenbank	160
	8.2.6 Systemdokumente	162
	8.2.7 Workflow-Server	163
9	Realisierung	165
9.1	Die Architektur des Workflow-Management-Systems CSE Workflow.	165
9.2	Das prototypische System Workflow Memory Information System (WoMIS)	166
9.2.1	Funktionalität	166
9.2.2	Architektur und Implementierung	170
9.3	Das prototypische System KontextNavigator	172
9.3.1	Funktionalität	172
9.3.2	Benutzerinteraktion	173
9.3.3	Architektur	179
9.3.4	KontextNavigator-Datenbank	186
9.3.5	Implementierungsaspekte	189
E	Abschluss	191
10	Zusammenfassung und Ausblick	193
F	Literatur und Verzeichnisse	195
	Literatur	197
	Abbildungsverzeichnis	209
	Tabellenverzeichnis	211
	Index	213

A Einleitung

1 Einleitung und Aufgabenstellung

1 Einleitung und Aufgabenstellung

Workflow-Management-Systeme (WFMS) sind Softwaresysteme, die anhand eines definierten Prozessmodells die Bearbeitung von Aufgaben durch Personen, die bestimmte Rollen innehaben oder bestimmten Organisationseinheiten zugeordnet sind, unter Bereitstellung der relevanten Informationsobjekte unterstützen. Obwohl WFMS in den 90er Jahren sowohl in der wissenschaftlichen Diskussion als auch im unternehmerischen Umfeld als Zukunftstechnologie für die informationstechnische Unterstützung von Geschäftsprozessen angesehen wurden, konnten diese Ansprüche bisher nur in geringem Maße gehalten werden: Die neue Technologie wurde zwar in den Unternehmen interessiert aufgenommen, Projekte nennenswerter Größenordnung wurden aber nur in vergleichsweise geringem Ausmaß durchgeführt. Darüber hinaus haben in der Vergangenheit eine Reihe von Workflow-Management-Projekten durch negative Erfahrungen von sich Reden gemacht, die zu einer weiteren Verunsicherung von potentiellen Anwendern führten.

Einer der Hauptgründe für die bisher mangelnde Akzeptanz von WFMS liegt in der eingeschränkten Klasse von Prozessen, die die überwiegende Anzahl kommerzieller WFMS unterstützen können: Das Prinzip des Workflow Managements liegt in einer Unterstützung von standardisierten, möglichst für jeden Fall gleich durchzuführenden Prozessen, die durch ein Prozessmodell festgelegt werden. Die unternehmerische Praxis zeigt jedoch, dass diese stark strukturierten Prozesse zwar in bestimmten Bereichen und Branchen existieren, oftmals aber weitergehende Anforderungen an eine flexible Ausführung und kontextbezogene Wissens- bzw. Informationsversorgung der Prozessbearbeiter bestehen, als diese von heutiger Workflow-Management-Technologie umgesetzt werden können.

In den letzten Jahren ist das systematische Management von Wissen sowohl in der wissenschaftlichen Welt als auch in Unternehmen und Verwaltungen verstärkt in das Zentrum der Betrachtung gerückt. Dies ist nicht zuletzt darauf zurückzuführen, dass ein hoher Anteil der Geschäftsprozesse der Zukunft „wissensintensiv“ sein wird. Diese wissensintensiven Geschäftsprozesse zeichnen sich durch flexible Wissensbedarfe und -ergebnisse aus, sie enthalten Aktivitäten, die hohe Kommunikations- und Kooperationsbedarfe einschließen oder komplexe Entscheidungen beinhalten, und sie erfordern einen geschäftsfallübergreifenden Wissenstransfer, d.h. die Weitergabe von Wissen, Erfahrungen, Gründen für Entscheidungen, etc. über die Grenzen des einzelnen Geschäftsfalls hinweg. Eine informationstechnische Unterstützung wissensintensiver Prozesse kann nur gelingen, wenn Workflow Management und Wissensmanagement nicht isoliert betrachtet werden, sondern die relevanten technologischen Aspekte aus den beiden Bereichen geeignet integriert werden.

Einerseits werden zur informationstechnischen Unterstützung von Wissensmanagement zur Zeit Organizational Memory Information Systems (OMIS) sehr stark diskutiert. Dabei handelt es sich um Systeme, die den Anspruch haben, das gesamte unternehmensrelevante Wissen (oder zumindest einen signifikanten Anteil) aus unterschiedlichen Wissens- oder Informationsquellen zu verwalten und bedarfsgerecht zur Verfügung zu stellen. Im Allgemeinen bieten diese Systeme aber keine Integration in die Prozessbearbeitung, sodass im Hinblick auf eine bedarfsgerechte Verfügbarkeit von Wissen Defizite bestehen.

WFMS sind andererseits für eine Unterstützung wissensintensiver Prozesse zu unflexibel: Zumeist werden die dem Bearbeiter bei der Prozessdurchführung zur Verfügung stehenden Informationen zum Modellierungszeitpunkt fest definiert. Eine flexible Bereitstellung prozessbezogenen Wissens in Abhängigkeit von der konkreten Bearbeitungssituation oder der Charakteristik des konkreten Geschäftsfalls ist lediglich in sehr eingeschränktem Umfang möglich. Allerdings bieten WFMS als Basismechanismus die bei OMIS fehlende Modellierung und Ausführung von Prozessen.

Unter prozessbezogenem Wissen verstehen wir das Wissen, das für die Bearbeitung von (Geschäfts-)Prozessen relevant ist. Dabei unterscheiden wir primäres und sekundäres Wissen: Primäres Wissen ist für die Bearbeitung jeden Geschäftsfalls relevant und wird vom Modellierer im Prozessmodell festgelegt. Zu primärem Wissen gehören somit u.a. Informationen über Aktivitäten und die Reihenfolge ihrer Bearbeitung, die zugehörigen Bearbeiter und Werkzeuge und die Dokumente, die jeweils in einer Aktivität benötigt oder bearbeitet werden. Darüber hinaus existiert mit sekundärem Wissen weiteres prozessbezogenes Wissen, das für die Bearbeitung von Aktivitäten in Prozessen wichtig ist und dessen Relevanz oftmals erst situativ bei einer konkreten Prozessbearbeitung vom Bearbeiter festgestellt werden kann. Daher kann sekundäres Wissen, wie z.B. Gründe für Entscheidungen, Erfahrungen, Vorgaben bzgl. der Bearbeitung oder weitere Hintergrundinformationen, nicht sinnvollerweise im Prozessmodell definiert werden.

Während nun primäres Wissen von heutigen WFMS in geeigneter Weise unterstützt wird, fehlen Möglichkeiten zur Verwaltung sekundären Wissens. Der in dieser Arbeit vorgestellte Ansatz verfolgt daher die Konzeption eines prozessbezogenen OMIS, d.h. eines Systems, das das für die Prozessdurchführung relevante sekundäre Wissen verwaltet, und dessen Integration in ein WFMS. Durch die Integration dieser beiden Technologien wird eine Unterstützung wissensintensiver Prozesse ermöglicht und das Anwendungsspektrum von Workflow-Management-Technologie wesentlich erweitert. In Bezug auf die Verwaltung prozessbezogenen Wissens folgt das Konzept den im Weiteren skizzierten Kernanforderungen:

- Verwaltung sekundären Wissens unterschiedlicher Funktion und Charakteristik:

Um ein möglichst breites Spektrum prozessbezogenen Wissens zu repräsentieren, sind sowohl die vom WFMS verwalteten Prozessmodell- und Laufzeitinformationen als auch das außerhalb des WFMS in Form von Dokumenten oder durch Informationssysteme repräsentierte Wissen zu betrachten. Schließlich ist zur Bearbeitung von Prozessen insbesondere auch das implizite Wissen der Bearbeiter, also nicht dokumentierte Erfahrungen oder Gründe für Entscheidungen, von hohem Interesse.

Die Relevanz sekundärem Wissens ergibt sich oftmals erst aus der Charakteristik der konkreten Bearbeitungssituation heraus. Aspekte, die hier insbesondere betrachtet werden müssen, sind die Fragen, wann das Wissen benötigt wird, ob das Wissen nur unter bestimmten Umständen von Interesse ist, also z.B. in Abhängigkeit von Eigenschaften des Geschäftsfalls, oder ob das Wissen zu bestimmten Ereignissen der Prozessbearbeitung aktiv angeboten werden soll. Nur wenn die Bereitstellung von Wissen von solchen prozessbezogenen Charakteristika abhängig gemacht werden kann, kann von einer echten wissensbezogenen Unterstützung der Prozessbearbeitung gesprochen werden.

- Strukturierung sekundären Wissens anhand des Bearbeitungskontexts:

Gerade bei komplexen wissensintensiven Prozessen kann jeweils eine erhebliche Menge sekundären Wissens zur Durchführung relevant sein. Dabei sind diese Wissensbestände aber im Allgemeinen lediglich in einem bestimmten Bearbeitungskontext wesentlich. Dieser Bearbeitungskontext kann z.B. durch die Durchführung einer bestimmten Aktivität oder durch die Nutzung eines bestimmten Werkzeugs charakterisiert sein. Daher ist eine kontextorientierte Unterstützung gefordert, die dem Bearbeiter durch eine geeignete Strukturierung das jeweils relevante Wissen präsentiert.

- Kooperativer Aufbau und Bewahrung sekundären Wissens:

Da sich bei wissensintensiven Prozessen flexible Wissensbedarfe ergeben, die oftmals erst zur Laufzeit durch die Bearbeiter identifiziert werden, muss der Aufbau und die Bewahrung sekundären Wissens in kooperativer Weise erfolgen. Dazu sind neben informationstechnischen Konzepten organisatorische Regelungen notwendig, die aufzeigen, welche Personen für die Bewahrung welcher Wissensbestände verantwortlich sind.

Zur Umsetzung der beschriebenen Anforderungen werden wir das Konzept der „Kontexte“ vorstellen. Kontexte strukturieren sekundäres Wissen anhand der Bearbeitungssituation und sind für den Bearbeiter integriert in die Prozessdurchführung durch den geeignet zu erweitern den WFMS-Client zugreifbar. Der Bearbeiter erhält also über das im Prozessmodell definierte primäre Wissen hinaus Zugriff auf weiteres sekundäres Wissen, das in seiner konkreten Bearbeitungssituation für ihn relevant ist. Darüber hinaus ist es ihm möglich, eigenes Wissen repräsentiert durch Dokumente in Kontexten abzulegen und somit zu einem geschäftsfallübergreifenden Wissenstransfer beizutragen. Die durch die Kontexte vorgegebene Struktur liefert insbesondere einen wesentlichen Beitrag für die Frage der geeigneten Wissensbewahrung.

Das Konzept beinhaltet als Kernelemente erstens eine Aufstellung relevanter Kontexttypen und damit die grundsätzliche Ordnungsstruktur zur Verwaltung sekundären Wissens und zweitens ein Informationsmodell für in Kontexten verwaltete Dokumente. Ein wesentlicher Vorteil des Kontextkonzepts besteht darin, dass die Informationen zum Aufbau der Ordnungsstruktur direkt vom WFMS bereitgestellt werden können. Damit entfallen zeitaufwändige und komplexe Aktivitäten zur Erstellung und Wartung der Ordnungsstruktur, wie sie oftmals bei inhaltsorientierten Strukturierungen durch Ontologien, Thesauri, etc. auftreten.

Die zentrale Anforderung in Bezug auf die organisatorische Einbettung von Kontexten besteht in der geeigneten Integration sekundären Wissens und damit des Kontextkonzepts in den Prozesslebenszyklus. Wir werden daher darstellen, welche Aufgaben unterschiedliche Gruppen bzw. Rollen, wie z.B. Modellierer, Kontextverantwortlicher oder Bearbeiter, in den einzelnen Phasen des Prozesslebenszyklus in Bezug auf die Verwaltung von Kontexten wahrnehmen. Dabei liegt der Schwerpunkt der Betrachtung auf der Fragestellung, mit welchen insbesondere organisatorischen Mitteln die kooperative Erstellung und Bewahrung von sekundärem Wissen sichergestellt werden kann.

Für eine informationstechnische Umsetzung des Kontextkonzepts sind WFMS in verschiedener Hinsicht zu erweitern. Wir werden daher eine Systemarchitektur für in diesem Sinne erweiterte WFMS vorstellen und den Beitrag der einzelnen Komponenten zur Verwaltung der einzelnen Aspekte prozessbezogenen Wissens erläutern. Dabei stellen wir insbesondere einen Vorschlag zur Präsentation von Kontexten innerhalb des Workflow-Clients vor.

Die Kernaspekte des erarbeiteten Konzepts wurden in zwei prototypischen Erweiterungen von WFMS umgesetzt. Dabei zeigt das im Rahmen des Verbundprojekts MOVE¹ realisierte System WoMIS (Workflow Memory Information System) demonstratorhaft das Prinzip der Kontexte und ihre Integration in ein WFMS in einer frühen Version. Der Prototyp KontextNavigator geht in seiner Funktionalität darüber hinaus und setzt die in dieser Arbeit beschriebenen Konzepte weitgehend um.

Die Arbeit ist wie folgt aufgebaut: In Teil B stellen wir zunächst die relevanten Grundlagen für diese Arbeit vor. Dazu gibt Kapitel 2 einen Überblick über die relevanten Aspekte aus den Bereichen Workflow Management und Wissensmanagement. In Kapitel 3 werden sowohl existierende Ansätze zur Integration von Workflow Management und Wissensmanagement diskutiert als auch eine Bewertung der Stärken und Defizite dieser Konzepte vorgenommen. Anschließend erläutert Kapitel 4 unsere Sicht auf den Begriff des wissensintensiven Prozesses anhand von Kriterien zur Charakterisierung dieser Prozessklasse. Außerdem stellen wir den Prozess „Contract Management“ vor, einen wissensintensiven Prozess zur Anbahnung und Verwaltung von Verträgen mit Sonderkonditionen bei einem Kurier- und Expressdienstleister, den wir im Rahmen des Verbundprojekts MOVE über mehrere Jahre analysiert und begleitet haben und den wir in dieser Arbeit durchgängig als Beispiel für einen wissensintensiven Prozess verwenden werden. In Teil C stellen wir unser Konzept zur Erweiterung von WFMS zur Unterstützung wissensintensiver Prozesse vor. Dazu diskutieren wir in Kapitel 5 zunächst Anforderungen an die Konzeption eines prozessbezogenen OMIS und dessen Integration in ein WFMS. In Kapitel 6 stellen wir das Konzept der Kontexte vor, mit dessen Hilfe die beschriebenen Anforderungen umgesetzt werden. Darauf aufbauend schlägt Kapitel 7 eine Methodik zur Integration der Verwaltung von Kontexten in den Prozesslebenszyklus vor. Teil D erläutert die informationstechnische Umsetzung des Konzepts und stellt entsprechende prototypische Realisierungen dar. Zunächst wird in Kapitel 8 anhand einer Systemarchitektur die informationstechnische Umsetzung von Kontexten als Erweiterung bestehender WFMS erläutert. In Kapitel 9 stellen wir schließlich zwei prototypische Realisierungen vor, die die Umsetzung der Kernaspekte des dargestellten Konzepts verdeutlichen. Schließlich wird in Teil E (Kapitel 10) eine Zusammenfassung der Arbeit und ein Ausblick gegeben.

1. Das Projekt MOVE wurde durch das Bundesministerium für Bildung, Wissenschaft, Forschung und Technologie unter der Fördernummer 01 HB 9404/6 gefördert. Die Ergebnisse des Projekts MOVE wurden in [HSW98a, HSW98b, HSW99, HSW01] veröffentlicht.

B Grundlagen für eine Unterstützung wissensintensiver Prozesse durch Workflow-Management-Systeme

- 2 Grundlagen des Workflow Managements und des Wissensmanagements
- 3 Ansätze zur Integration von Workflow Management und Wissensmanagement
- 4 Wissensintensive Prozesse

2 Grundlagen des Workflow Managements und des Wissensmanagements

In diesem Kapitel stellen wir die für unseren Ansatz der Unterstützung wissensintensiver Prozesse durch Workflow-Management-Systeme relevanten Grundlagen vor. Dazu werden die für diese Arbeit relevanten Aspekte aus den Bereichen Workflow Management, Wissensmanagement und Dokumentenmodellierung vorgestellt.

2.1 Workflow-Management-Systeme

Workflow Management Systeme (WFMS) wurden in den 90er Jahren sowohl in der wissenschaftlichen Diskussion als auch im unternehmerischen Umfeld als Zukunftstechnologie für die informationstechnische Unterstützung von Geschäftsprozessen angesehen. Allerdings konnten die Hoffnungen, die an die Technologie geknüpft wurden, nur zu einem geringem Teil erfüllt werden, da Projekte nennenswerter Größenordnung nur in vergleichsweise dürftigem Ausmaß durchgeführt wurden und eine Reihe von Negativerfahrungen beim Einsatz von WFMS bekannt wurden [GS98]. Dies liegt nicht zuletzt daran, dass die überwiegende Anzahl kommerzieller WFMS lediglich eine eingeschränkte Klasse von Prozessen der unternehmerischen Praxis unterstützen kann und insbesondere Anforderungen an eine flexible Wissens- bzw. Informationsversorgung der Mitarbeiter nur unzureichend umgesetzt werden können. Gleichwohl handelt es sich, wie in dieser Arbeit weiter ausgeführt wird, bei WFMS um eine Kernkomponente in einer Architektur zur Unterstützung von wissensintensiven Prozessen. Daher stellen wir in diesem Abschnitt sowohl die Grundlagen als auch die Defizite von WFMS vor.

Abschnitt 2.1.1 erläutert zunächst die grundsätzlichen Eigenschaften von WFMS sowie die wesentlichen Begriffe des Workflow Managements und gibt einen kurzen Überblick über die abstrakte Architektur von WFMS. Das Management von Geschäftsprozessen durch WFMS ist durch ein zyklisches Vorgehen geprägt. Diesen Prozesslebenszyklus stellen wir in Abschnitt 2.1.2 vor. Daraufhin erläutern wir die Möglichkeiten zur Verwaltung von prozessbezogenem Wissen und Informationen durch heutige WFMS (Abschnitt 2.1.3). Wir werden in späteren Kapiteln darauf Bezug nehmen und zeigen, dass diese existierenden Funktionalitäten von WFMS zur Unterstützung wissensintensiver Prozesse nicht ausreichend sind. Schließlich beschreiben wir das Einsatzspektrum heutiger WFMS anhand einer Charakterisierung workflow-gerechter Prozesse (Abschnitt 2.1.4). Wir werden diese Charakterisierung in Kapitel 4 aufnehmen und die Klasse der wissensintensiven Prozesse als eine praxisorientierte Erweiterung des Einsatzspektrums von WFMS vorstellen.

2.1.1 Überblick und Begriffsbestimmung

Workflow Management umfasst die Modellierung und Durchführung von Geschäftsprozessen. Das Ziel des Workflow Managements besteht grundsätzlich darin, im Rahmen eines Prozesses die Bearbeitung von Aufgaben durch Personen, die bestimmte Rollen innehaben oder bestimmten Organisationseinheiten zugeordnet sind, unter Verwendung von Informationsob-

jekten und Werkzeugen zu unterstützen. Die entsprechenden Prozesseigenschaften werden dazu bei der Modellierung durch ein Prozessmodell und ggf. ein Organisationsmodell beschrieben. WFMS koordinieren auf der Basis dieser Beschreibung die Durchführung von Aufgaben durch verschiedene Personen, indem sie die entsprechenden Personen über anstehende Aufgaben informieren, relevante Informationsobjekte bereitstellen und entsprechende Werkzeuge zur Bearbeitung der Informationsobjekte integrieren. WFMS sind daher Basissysteme, die keine fachliche Funktionalität beinhalten, und erst durch eine entsprechende Konfiguration produktiv eingesetzt werden können.

Im Folgenden werden die für die Betrachtung des Workflow Managements wesentlichen Begriffe definiert. In der Literatur hat sich bisher keine einheitlich verwendete Terminologie herausgebildet. Insbesondere die Begriffe Prozess, Workflow oder Vorgang werden in der Literatur oftmals vollkommen unterschiedlich verwendet. Normierungsaktivitäten wurden zwar durch die Workflow Management Coalition (WfMC) durchgeführt, ein entsprechender Vorschlag für eine Terminologie zur Beschreibung von Workflow Management [WfMC99a] konnte sich allerdings nicht durchsetzen¹. Daher werden im Weiteren eigene Definitionen für die in dieser Arbeit relevanten Begriffe gegeben. Als Basis dienten dazu insbesondere die Begriffsbildungen in [Lind99] und [Stri98].

Definition 1: Prozess

„Ein Prozess besteht aus entweder einer Aktivität oder einer Menge von verknüpften Aktivitäten oder Prozessen (diese werden als Subprozesse bezeichnet). Ein Prozess dient zur Erreichung eines bestimmten Unternehmensziels.“

Die Beschreibung eines Prozesses erfolgt durch ein zugehöriges Prozessmodell. Prozesse können mehrstufig hierarchisch aufgebaut sein, indem sie Subprozesse enthalten.

Definition 2: Aktivität

„Bei einer Aktivität handelt es sich um eine Menge von Arbeitsschritten, die von einem Bearbeiter ausgeführt werden. Eine Aktivität benötigt und bearbeitet dazu eine Menge von Dokumenten und wird von einem Bearbeiter oder automatisch unter Verwendung von Werkzeugen durchgeführt.“

Definition 3: Dokument

„Ein Dokument ist eine Menge von rechnergestützten Informationen, die Rahmen eines Prozesses benötigt oder bearbeitet werden.“

Der Begriff Dokument wird hier bewusst weit gefasst, um alle prozessbezogenen Informationen einheitlich beschreiben zu können. Damit umfasst der Dokumentbegriff sowohl Informationen, die in Form von klassischen elektronische Dokumenten, wie z.B. Office-Dokumenten, vorliegen, als auch Daten, wie z.B. Einträgen in Datenbanken. Mit Hilfe dieser Begriffsbildung

1. Weitere Ansätze zur Normierung der Begriffswelt, die allerdings auch nicht den gewünschten Erfolg erzielten, wurden vom Deutschen Institut für Normung [DIN96], dem Arbeitskreis der Gesellschaft für Informatik (GI) „Modellierung und Ausführung von Workflows“ [JBS97], im Projekt Vorgangsbearbeitungs-Teleservices (VORTEL) [BDFL96] und von der Process Interchange Format-Arbeitsgruppe [LGJM96] entwickelt.

soll einerseits die Verwendung des Begriffs „Information“ vermieden werden, andererseits sollen nicht fortwährend bei der Beschreibung von prozessbezogenen Informationen die Begriffe Daten und Dokumente unterschieden werden müssen.

Definition 4: Prozessmodelldokument

„Ein Prozessmodelldokument ist ein Dokument, das als Teil einer Prozessbeschreibung in einem zugehörigen Prozessmodell enthalten ist.“

In vielen Fällen kann eine Zuordnung von Dokumenten zu Prozessen oder zu einzelnen Aktivitäten bereits zum Modellierungszeitpunkt im Prozessmodell festgelegt werden. Diese Dokumente nennen wir im Folgenden Prozessmodelldokumente. Wir werden in dieser Arbeit allerdings deutlich machen, dass in vielen Fällen zur Bearbeitung von Prozessen weitere Dokumente notwendig sind, die nicht durch das Prozessmodell beschrieben werden können.

Definition 5: Bearbeiter

„Ein Bearbeiter ist ein menschlicher Ausführer einer Aktivität eines Prozesses.“

Definition 6: Workflow

„Ein Workflow ist ein Prozess, der von einem Workflow-Management-System zum Ablauf gebracht werden kann.“

Die Definition eines Prozesses (unter Bezugnahme auf die Begriffe Aktivität und Dokument) wurde so gewählt, dass es sich bei einem Prozess um einen formal beschriebenen, durch ein Prozessmodell repräsentierten Ablauf in einem Unternehmen handelt. Daher bestehen die Unterschiede bei der Begriffsbildung zwischen Prozess und Workflow lediglich darin, dass ein Workflow stets durch ein WFMS zur Ausführung gebracht werden kann, während diese Eigenschaft nicht notwendigerweise für Prozesse gilt. Wir verwenden in dieser Arbeit zumeist den Oberbegriff Prozess. Lediglich, wenn die Ausführbarkeit von Prozessen durch WFMS besonders hervorgehoben werden soll oder existierende Ansätze oder Systeme aus dem Bereich des Workflow Managements beschrieben werden, wird teilweise der Begriff Workflow verwendet.

Definition 7: Geschäftsfall

„Ein Geschäftsfall ist die Repräsentation einer einzelnen Ausführung eines Prozesses.“

Ein Geschäftsfall ist damit die Summe der Zustände der erzeugten oder bearbeitenden Dokumente und der Historie der durchgeführten bzw. sich in Durchführung befindlichen Aktivitäten.

Definition 8: Workflow-Management-System

„Ein Workflow-Management-System (WFMS) ist ein Softwaresystem, das Workflows gemäß der entsprechenden Prozessmodelle zum Ablauf bringt.“

WFMS unterstützen die Zusammenarbeit von Personen durch die kontrollierte Ausführung von Arbeitsabläufen und können daher in den Bereich der Computer Supported Cooperative Work (CSCW) eingeordnet werden. Rechnergestützte Gruppenarbeit wird ferner durch Groupware unterstützt, wobei bestimmte Systeme, wie z.B. Microsoft Exchange oder Lotus Domino, ebenfalls Funktionalitäten anbieten, die zu einer Steuerung von Arbeitsabläufen genutzt werden können, ohne dass diese Systeme aber die Definition eines expliziten Prozessmodells

ermöglichen. Die hier durchgeführte Abgrenzung von WFMS zu verwandten Systemklassen im Bereich der CSCW wird daher an der Existenz eines Prozessmodells festgemacht. Nur wenn eine eindeutige Trennung zwischen Prozessmodell und Ausführungskomponente vorliegt, sprechen wir von einem Workflow-Management-System.

Definition 9: Workflow-Management-Anwendung

„Eine Workflow-Management-Anwendung besteht aus mindestens drei voneinander getrennten Komponenten, nämlich einem Workflow-Management-System, einem durch das Workflow-Management-System interpretierbaren Prozessmodell sowie aus den im Prozessmodell definierten Werkzeugen.“

Zur weiteren Verdeutlichung der Unterstützung von Prozessen durch Workflow Management und der einzelnen Komponenten eines WFMS wird im Folgenden die abstrakte Architektur eines WFMS anhand des Referenzmodells der Workflow Management Coalition (WfMC) [WfMC95] vorgestellt (Abbildung 1). Das Referenzmodell der WfMC dient als allgemeines Architekturmodell für verschiedene Arten von WFMS und soll insbesondere die Standardisierung von Schnittstellen zwischen Komponenten unterschiedlicher Hersteller unterstützen.

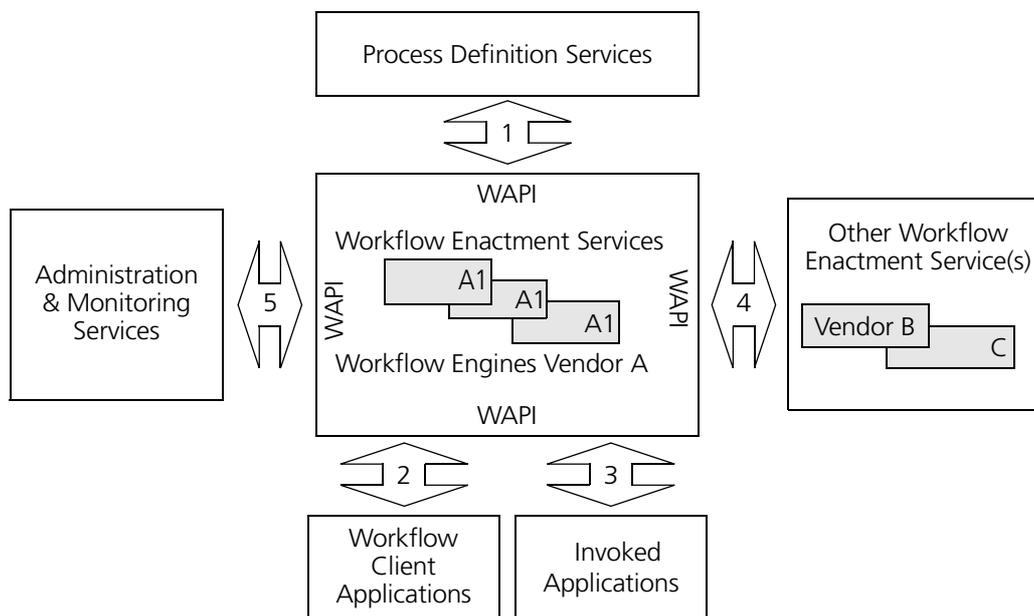


Abbildung 1: Referenzmodell der Workflow Management Coalition [WfMC95]

Die zentrale Komponente im Architekturvorschlag der WfMC ist der Workflow Enactment Service (WES). Der WES ist für die Erzeugung, Verwaltung und Ausführung von Geschäftsfällen verantwortlich und kommuniziert mit den anderen Komponenten des WFMS über das Workflow Application Programmer Interface (WAPI). Im Folgenden soll nur ein Überblick über die dargestellten Komponenten eines WFMS gegeben werden, ohne dass auf die Spezifika der einzelnen WAPI-Schnittstellen eingegangen wird.

Unter Process Definition Services (WAPI-Schnittstelle 1) werden Dienste zur Manipulation von Prozessbeschreibungen verstanden. Die meisten kommerziellen WFMS bieten hier einen Editor an, mit dem der Benutzer die Prozessmodelle in einer graphischen Notation definieren kann.

Workflow Client Applications (WAPI-Schnittstelle 2) dienen zum Informationsaustausch zwischen dem WES und Bearbeitern während der Durchführung von Workflows. Workflow Client Applications verwalten für jeden Bearbeiter eine Arbeitsliste, die alle Aktivitäten enthält, für die der entsprechende Bearbeiter in ggf. mehreren Workflows augenblicklich zuständig ist. Der Bearbeiter kann aus dieser Arbeitsliste Aktivitäten auswählen und diese starten, terminieren, etc.

Zur Unterstützung der Bearbeitung von Aufgaben können im Prozessmodell Werkzeuge festgelegt sein. Diese Werkzeuge werden durch Applikationen (WAPI-Schnittstelle 3) implementiert, die nicht Teil des WFMS sind und durch den WES aufgerufen und mit den entsprechenden Prozessmodelldokumenten versorgt werden.

Die WAPI-Schnittstelle 4 bietet die Möglichkeit der Kommunikation zwischen Workflow Enaction Services ggf. unterschiedlicher Hersteller. Hiermit ist es möglich, dass laufende Workflows mit anderen Workflows kommunizieren.

Schließlich werden unter Administration and Monitoring Tools alle Administrationsdienste eines WFMS zusammengefasst (WAPI-Schnittstelle 5). Die entsprechenden Komponenten sollen die Überwachung aller laufenden Workflows sicherstellen, so z.B. bei Fehlern oder Sonderfällen in den laufenden Workflow eingegriffen werden kann. Zusätzlich verwalten die Administration and Monitoring Tools Informationen über die Benutzer des Systems.

2.1.2 Der Prozesslebenszyklus

Das Management von Prozessen durch WFMS ist durch ein zyklisches Vorgehen charakterisiert. Dieser Prozesslebenszyklus (hier gemäß [JHS98] dargestellt, ähnliche Ansätze finden sich in [Ders96, DGS95 und Heil94]) ist grundlegend für das Verständnis der Erstellung und kontinuierlichen Verbesserung von Workflow-Management-Anwendungen und soll im Folgenden kurz vorgestellt werden¹. Wir werden uns später bei der Diskussion von Anforderungen und eines Konzepts bzgl. der Unterstützung wissensintensiver Prozesse durch WFMS auf die hier vorgestellten Begriffe beziehen.

Der zyklische Prozess (Abbildung 2) beginnt mit der Phase der Erhebung. In dieser Phase wird der Ist-Zustand der relevanten Prozesse durch z.B. Prozessstelleninterviews, moderierte Gruppendiskussionen oder die Analyse von Dokumenten erhoben.

1. Die Beschreibung der Erstellung und kontinuierlichen Verbesserung einer Workflow-Management-Anwendung anhand eines groben Phasenmodells ist für diese Arbeit ausreichend. Wir haben an anderer Stelle [WGHS99, DGS98] ein detaillierteres Referenzmodell für das Vorgehen in Workflow-Management-Projekten dargestellt.

Im Rahmen der Modellierung werden die erhobenen Prozesse in einer formalen Darstellung, dem Prozessmodell, abgebildet. Dabei werden die durchzuführenden Aktivitäten, ihre zeitlichen Bedingungen zueinander, die benötigten Dokumente und Werkzeuge und die beteiligten Organisationseinheiten und Rollen definiert. Eine Unterscheidung von einerseits Workflow-Modellen, die direkt von einem WFMS ausführbar sind, und andererseits Geschäftsprozessmodellen, deren Inhalte eher durch betrieblich-organisatorische Zusammenhänge festgelegt sind und die nicht direkt durch ein WFMS interpretiert werden können, bringt für diese Arbeit keinen Nutzen. Daher gehen wir im Folgenden von einem integrierten Ansatz [SHW97] zur Entwicklung von Workflow-Management-Anwendungen aus. Bei integrierten Ansätzen wird keine Trennung zwischen einem Geschäftsprozessmodell und einem Workflow-Modell im oben beschriebenen Sinne vorgenommen, sondern alle Informationen, die in den einzelnen Phasen des Prozesslebenszyklus produziert oder benötigt werden, werden in einem zentralen Prozessmodell gehalten.

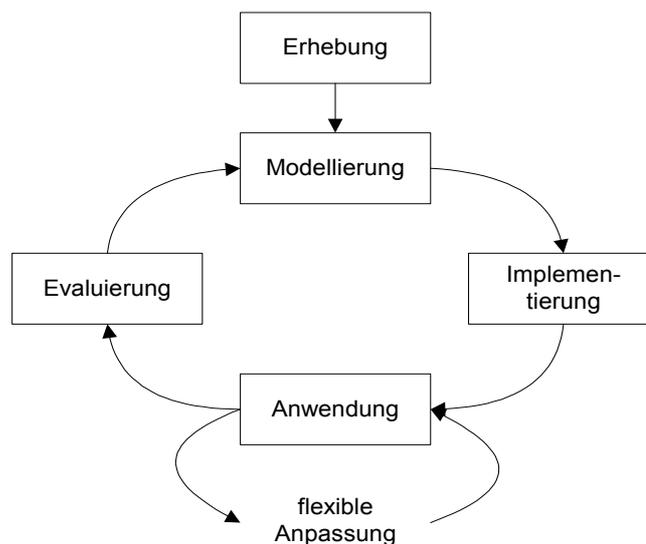


Abbildung 2: Der Prozesslebenszyklus [JHS98]

In der Phase der Implementierung wird die Workflow-Management-Anwendung erstellt. Dazu wird einerseits die technische Workflow-Management-Infrastruktur aufgebaut und andererseits das Prozessmodell um die Aspekte erweitert, die für eine Durchführungsunterstützung durch das WFMS notwendig sind. Dies impliziert z.B. die Implementierung von Prozessmodell-dokumenten und die Integration von externen Anwendungen über geeignete Schnittstellen.

Die beiden folgenden Phasen der Anwendung und der flexiblen Anpassung sind gemeinsam zu betrachten: Während der Anwendung führt das WFMS die implementierten Prozesse aus und koordiniert dabei die einzelnen Bearbeiter, steuert die Durchführung von Aktivitäten und stellt dazu die entsprechenden Dokumente und Werkzeuge bereit. Treten Ausnahmen oder Sonderfälle auf, besteht für den Bearbeiter oftmals die Notwendigkeit, vom Prozessmodell abzuweichen, also den aktuellen Geschäftsfall flexibel an die besonderen Gegebenheiten anzupassen.

In der Evaluierungsphase werden die während der Anwendung gesammelten Informationen ausgewertet. Diese Auswertung bietet die Basis für den Vergleich dieser Informationen (z.B. durchschnittliche Laufzeit eines Geschäftsfalls) mit den festgelegten Zielen (z.B. geschätzte durchschnittliche Laufzeit eines Geschäftsfalls). Zusätzlich sollten die Bearbeiter bei der Ausführung von Prozessen die Möglichkeit haben, Feedback und explizite Verbesserungsvorschläge abzugeben [JR99]. Diese gesammelten Informationen bieten eine Grundlage für eine Modifikation des Prozessmodells in einer folgenden Phase und damit zur kontinuierlichen Verbesserung des Prozesses.

Bei der Einführung und dem Betrieb einer Workflow-Management-Anwendung sind mehrere Rollen zu unterscheiden. Prozessmodelle werden von Modellierern erstellt. Die Implementierung der Workflow-Management-Anwendungen wird von Programmierern realisiert. Bearbeiter führen die implementierten Workflows durch, während Administratoren die Ausführung überwachen und für die Evaluierung und kontinuierliche Verbesserung von Prozessen verantwortlich sind. Oftmals übernehmen Personen in Workflow-Management-Projekten jeweils mehrere Rollen, indem z.B. der Modellierer ebenfalls Teile der Implementierung der Workflow-Management-Anwendung durchführt.

Es ist von entscheidender Bedeutung für den Projekterfolg, dass das oben beschriebene Vorgehen unter kontinuierlicher Beteiligung der betroffenen Mitarbeiter durchgeführt wird. Bereits bei der initialen Phase der Erhebung als auch in den folgenden Phasen ist darauf zu achten, dass nicht nur die entwickelten Konzepte regelmäßig den Mitarbeitern präsentiert werden, sondern auch die Verbesserungswünsche und -vorstellungen der Mitarbeiter in die Konzepte einfließen [HGH98]. Dieser partizipative Ansatz äußert sich auch in der expliziten Darstellung der eigenverantwortlichen, flexiblen Anpassung des Geschäftsfalls durch die Bearbeiter im oben dargestellten Prozesslebenszyklus. Die in dieser Arbeit vorgestellte Erweiterung von WFMS zur Unterstützung wissensintensiver Prozesse wird grundsätzlich diesem partizipativen Ansatz folgen, da die Verantwortung für den Aufbau und die Bewahrung prozessbezogenen Wissens überwiegend bei den Bearbeitern der Prozesse liegen soll.

2.1.3 Verwaltung von prozessbezogenen Informationen in heutigen WFMS

Wie eingangs erläutert wurde, zielt diese Arbeit auf die Konzeption eines prozessbezogenen Organizational Memory Information Systems (OMIS - eine genauere Beschreibung von OMIS erfolgt in Abschnitt 2.2.3) und dessen Integration in ein WFMS zur Unterstützung wissensintensiver Prozesse ab. Dazu sind zunächst die grundsätzlichen Möglichkeiten und Defizite heutiger WFMS zur Verwaltung von prozessbezogenen Informationen zu betrachten. Unter prozessbezogenen Informationen verstehen wir die Informationen, wie z.B. Daten oder Dokumente, die während der Durchführung von Prozessen in Aktivitäten benötigt oder bearbeitet werden und unter der Kontrolle des WFMS stehen. Daher stellen wir in diesem Abschnitt mit Hilfe einer Klassifikation vor, welche Sicht auf prozessbezogene Informationen bei klassischen Workflow-Management-Ansätzen vorliegt, und skizzieren diesbezüglich die entsprechenden Eigenschaften einiger WFMS.

Zur Beschreibung des Informationsaspekts eines Workflow-Management-Ansatzes verwenden wir die Aspekte

- Beschreibungskonzepte der Modellierung,
- Zugriffsmethoden und
- Flexibilität.

In Bezug auf die Beschreibungskonzepte, die ein WFMS für die Modellierung des Informationsaspekts bietet, können datenorientierte und dokumentenorientierte Ansätze unterschieden werden. Bei einem datenorientierten Ansatz wird im Rahmen der Prozessmodellierung die Struktur der Daten definiert, und es werden Datentypen, Wertebereiche und Formate angegeben. Daraufhin legt der Modellierer den Datenfluss fest, der die geforderte Weiterleitung der Daten durch das WFMS im Rahmen des Prozesses beschreibt und nicht notwendigerweise mit dem Kontrollfluss identisch ist. Bei dokumentenorientierten WFMS werden prozessbezogene Informationen mittels Dokumenten zwischen den Bearbeitern ausgetauscht. Der Modellierer identifiziert dazu für einen Prozess eine Menge von Dokumenten (z.B. jeweils über die Angabe einer Datei), die jeweils als Template für die Prozessdokumente fungieren und beim Start eines konkreten Geschäftsfalls instanziiert werden. Darüber hinaus kann bei einigen Systemen definiert werden, welche Dokumente in welchen Aktivitäten für den Bearbeiter sichtbar sein sollen. Die interne Struktur der Dokumente wird anders als die Datenstrukturen beim datenorientierten Ansatz außerhalb des WFMS durch die entsprechenden Anwendungen zur Erzeugung der Dokumente (also z.B. durch klassische Office-Anwendungen) festgelegt.

Zum Zugriff auf die in der Modellierung festgelegten Daten oder Dokumente bieten WFMS entsprechende API-Funktionen an. Über diese Funktionen erhalten die in den Prozess integrierten Werkzeuge Zugriff auf die prozessbezogenen Informationen und können diese gemäß des Prozessmodells lesen bzw. verändern. Darüber hinaus bieten dokumentenorientierte Systeme durch den Workflow-Clients im Allgemeinen eine Umlaufmappe an, die dem Bearbeiter die für die aktuelle Aktivität relevanten Dokumente anzeigt. Der Bearbeiter kann nun - falls die entsprechenden Berechtigungen vorliegen - auf die Umlaufmappe und die darin enthaltenen Dokumente zugreifen, indem er z.B. Dokumente öffnet und modifiziert.

Der letzte Aspekt betrifft die Möglichkeiten, die ein WFMS in Bezug auf die flexible Behandlung des Informationsaspekts bietet. Hier können drei Alternativen unterschieden werden: Im ersten Fall lässt das WFMS keine Informationsflexibilität zu, d.h. in jedem Geschäftsfall eines Prozesses sind in jeder Aktivität genau die Daten oder Dokumente zugreifbar, die bei der Modellierung festgelegt wurden. Eine Abweichung vom Modell ist nicht möglich. Folglich eignen sich diese Systeme zur Unterstützung von Prozessen, bei denen die Menge der prozessbezogenen Informationen stark strukturiert ist und zusätzlich zum Modellierungszeitpunkt exakt angegeben werden kann, welche Aktivitäten auf welche Menge von Daten lesend bzw. schreibend zugreifen. Im zweiten Fall ist eine flexible Abweichung vom Modell auf Instanzebene erlaubt, d.h. der Bearbeiter kann den Informationsaspekt für einen laufenden Geschäftsfall modifizieren, indem er z.B. Datenstrukturen verändert oder dem Geschäftsfall neue Dokumente hinzufügt. Die Änderungen beziehen sich jedoch lediglich auf den aktuellen Geschäftsfall. Schließlich ermöglicht die dritte Alternative eine flexible Änderung des Prozessmodells zur Laufzeit. Änderungen bzgl. des Informationsaspekts wirken sich somit nicht nur auf den aktuellen Geschäftsfall, sondern auf den Prozess und alle zukünftigen Geschäftsfälle aus.

Die meisten datenorientierten WFMS lassen keine Flexibilität bzgl. des Informationsaspekts zu. Dagegen erlauben viele dokumentenorientierte Systeme die Ablage zusätzlicher Dokumente in der Umlaufmappe zur Laufzeit. Eine Umlaufmappe ist genau einem Geschäftsfall zugeordnet und wird nach Abschluss des Geschäftsfalls im Allgemeinen automatisch vom WFMS archiviert. Somit ist eine flexible Erweiterung der prozessbezogenen Informationen auf Instanzebene durch die Bearbeiter möglich. Im Folgenden stellen wir als weitere Erläuterung der genannten Aspekte die Behandlung prozessbezogener Informationen durch einige WFMS vor.

Der Ansatz der Workflow Management Coalition

Der Ansatz der WfMC [WfMC99a] unterscheidet drei Typen von Daten in WFMS, die für die Durchführung von Workflows relevant sind (Abbildung 3).

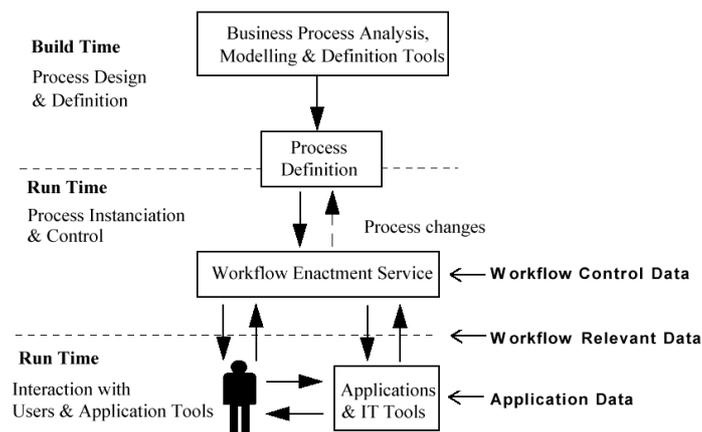


Abbildung 3: Typen von Daten in WFMS [WfMC99a]

Application Data bezeichnen Daten, die spezifisch für die in den Workflow eingebundenen Werkzeuge sind und die durch das WFMS nicht zugegriffen werden können. Workflow Control Data sind Daten, die das WFMS intern zur Steuerung der einzelnen Geschäftsfälle verwaltet und die für Bearbeiter oder Werkzeuge nicht zugänglich sind. Zu diesen Steuerdaten gehören z.B. Informationen über den Zustand jedes einzelnen Geschäftsfalls und jeder einzelnen Aktivität oder Recovery-Informationen zur Wiederherstellung des Systems nach einem Fehlerfall. Für die Betrachtung prozessbezogener Informationen im obigen Sinne sind allerdings Werkzeug-interne Daten (Application Data) und WFMS-interne Steuerdaten (Workflow Control Data) nicht relevant. Unter Workflow Relevant Data werden die Daten verstanden, die im Prozessmodell festgelegt sind und die für jeden Geschäftsfall während der Bearbeitung erzeugt und benutzt werden. Workflow Relevant Data können sowohl vom WFMS als auch von Werkzeugen manipuliert werden und dienen dem WFMS zusätzlich dazu, die korrekten Zustandsübergänge für einen Geschäftsfall festzulegen (z.B. kann nach Abschluss einer Aktivität die Entscheidung über die Folgeaktivität von den Workflow Relevant Data des jeweiligen Geschäftsfalls abhängen). In [WfMC99b] werden Workflow Relevant Data anhand von verschiedenen Attributen spezifiziert. Wichtig ist in diesem Zusammenhang, dass

Workflow Relevant Data typisiert sind, wobei sowohl einfache als auch zusammengesetzte Datentypen zugelassen werden. Es handelt sich somit bei dem Ansatz der WfMC um einen datenorientierten Ansatz. Möglichkeiten zur flexiblen Behandlung prozessbezogener Informationen durch den Bearbeiter werden nicht aufgezeigt.

Der Forschungsprototyp MOBILE

Jablonski & Bußler stellen in [JB96] ein Workflow-Metamodell für das prototypische WFMS MOBILE vor, das durch die Unterscheidung verschiedener Aspekte gekennzeichnet ist, die jeweils Teilfunktionalitäten von WFMS beinhalten: Dabei beschreibt der funktionale Aspekt die funktionalen Ausführungseinheiten eines Workflows (elementare und komposite Workflows). Der verhaltensbezogene Aspekt bestimmt den Kontrollfluss zwischen einzelnen Workflows. Der organisatorische Aspekt beinhaltet die Zuordnung von Organisationseinheiten, Personen oder Rollen zu Workflows. Die Einbindung von Werkzeugen in einen Workflow wird durch den operationalen Aspekt abgedeckt. Schließlich beschreibt der Informationsaspekt die in einem Workflow verwendeten Datentypen und den zwischen Workflows zu realisierenden Datenfluss.

Prozessbezogene Informationen werden in MOBILE im Rahmen des Informationsaspekts festgelegt. Zur Workflow-Definition stellt MOBILE eine Spezifikationssprache, die Mobile Specification Language (MSL), bereit. MSL erlaubt die Definition von Datentypen (einfache und zusammengesetzte), Workflow-Parametern, Workflow-Variablen (die lokalen Variablen eines Workflows) und Datenflussdefinitionen. Es handelt sich also um ein datenorientiertes System. Im Unterschied zum Ansatz der WfMC erlaubt MOBILE die flexible Modifikation prozessbezogener Informationen zur Laufzeit auf Instanz- und Typebene, indem der Bearbeiter z.B. Datentypen oder den Datenfluss verändert.

Das kommerzielle WFMS CSE Workflow¹

Bei CSE Workflow handelt es sich um ein WFMS, das eine starke Dokumentenorientierung aufweist. Ein Geschäftsfall besteht aus einer Umlaufmappe, die im wesentlichen Dokumente, die durch Dateien manifestiert sind, enthalten kann. Verweise auf Datenstrukturen sind zwar prinzipiell möglich, werden aber nicht durch Basismechanismen des Systems unterstützt. Der Weg einer Umlaufmappe wird durch das Prozessmodell vorgegeben. Dabei kann allerdings nicht festgelegt werden, welche Dokumente in welchen Aktivitäten sichtbar sein oder erzeugt werden sollen, es wird stets der gesamte Geschäftsfall mit allen Dokumenten von Aktivität zu Aktivität weitergeleitet.

Um eine Menge relevanter Dokumententypen vorzugeben, kann der Modellierer Vorlagen festlegen. Diese Vorlagen, wie z.B. Antrag, Bescheinigung, etc. werden mit einem entsprechenden Werkzeug zur Bearbeitung, wie z.B. MS Word oder MS Excel, verknüpft. Bei der Durchführung von Prozessen kann der Bearbeiter auf diese Vorlagen zugreifen, sie instanzieren und die erzeugten Dokumente in der Umlaufmappe des Geschäftsfalls ablegen. Danach können diese Dokumente bearbeitet, aber nicht mehr gelöscht werden. Darüber hinaus können die Bearbei-

1. CSE Workflow wird mittlerweile als Teil der SERfloware Suite von der SER AG vertrieben.

ter Zusatzinformationen zum Geschäftsfall in Form von Dateien, die nicht auf Vorlagen basieren, in der Umlaufmappe ablegen. Abbildung 4 zeigt diesbezüglich den Workdesk-Client von CSE Workflow, durch den Bearbeiter Geschäftsfälle und zugehörige Umlaufmappen manipulieren können, und stellt in der unteren Bildhälfte eine Umlaufmappe zu einem Geschäftsfall dar. Die gesamte Umlaufmappe mit allen Dokumentversionen wird bei Beendigung des Geschäftsfalls archiviert und kann zu einem späteren Zeitpunkt über Suchfunktionalitäten von CSE Workflow wieder aufgefunden werden. Funktionalitäten zum Austausch von Informationen zwischen unterschiedlichen Geschäftsfällen existieren nicht.

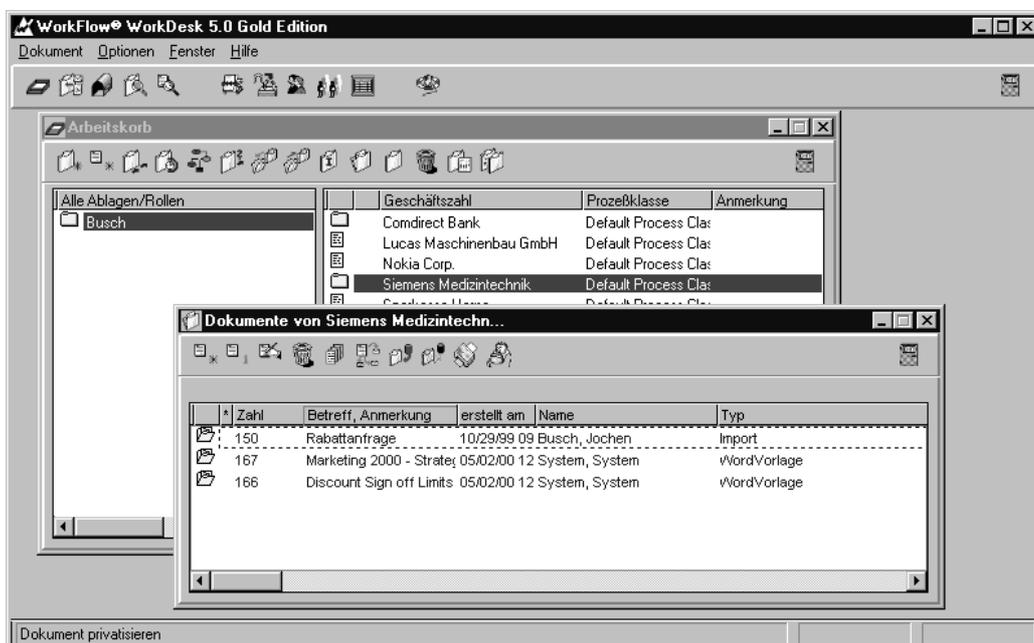


Abbildung 4: Workdesk-Client mit Umlaufmappe bei CSE Workflow

Weitere relevante dokumentenorientierte Systeme, die Informationsflexibilität auf der Instanzebene bieten und dem Bearbeiter eine elektronische Umlaufmappe bereitstellen, sind z.B. Compaq Link Works und COI BusinessFlow. Zu marktrelevanten datenorientierten WFMS zählen z.B. IBM MQSeries Workflow, COSA Workflow und Staffware 2000. Diese Systeme lassen keine Informationsflexibilität zu.

Aus den beschriebenen Eigenschaften heutiger WFMS ergibt sich folgendes Fazit: Bei heutigen WFMS werden im Informationsaspekt des Prozessmodells genau die prozessbezogenen Informationen (Daten oder Dokumente) festgelegt, die für die Bearbeitung eines jeden Geschäftsfalls notwendig sind. Informationsflexibilität, d.h. die Möglichkeit des Bearbeiters, die Menge der Informationsobjekte zur Laufzeit zu ändern und damit die besonderen Informationsbedarfe oder Ergebnisse des aktuellen Geschäftsfalls zu berücksichtigen, ist hauptsächlich bei dokumentenorientierten WFMS in eingeschränkter Weise z.B. durch das Hinzufügen weiterer Dokumente in die elektronische Umlaufmappe möglich. Datenorientierte WFMS-Produkte, die Informationsflexibilität bieten, konnten nicht identifiziert werden (bei dem flexiblen System MOBILE handelt es sich um einen Forschungsprototypen).

Wir werden später ausführlich darstellen (Kapitel 4 und Kapitel 5), dass zur Unterstützung wissensintensiver Prozesse Anforderungen an die Strukturierung und an die kooperative Erstellung und Bewahrung prozessbezogener Informationen gestellt werden, die über die beschriebenen Möglichkeiten von WFMS hinausgehen. Wir werden zeigen, dass es sinnvoll ist, neben den im Prozessmodell definierten Informationsobjekten weitere, „sekundäre“ Dokumente (Definition 14) zu verwalten, die einen eher unterstützenden Charakter für die Prozessbearbeitung besitzen und nicht sinnvollerweise in das Prozessmodell aufgenommen werden sollten. Die Möglichkeiten heutiger WFMS zur Unterstützung von Informationsflexibilität sind zur Verwaltung dieser sekundären Dokumente nicht ausreichend. Eine Integration von Workflow Management und Wissensmanagement stellt einen erfolgversprechenden Ansatz für eine Erweiterung der entsprechenden Funktionalitäten von WFMS dar. Wir werden im folgenden Kapitel daher existierende Arbeiten aus diesem Bereich diskutieren.

2.1.4 Einsatzspektrum heutiger Workflow-Management-Systeme

Aufbauend auf der bisher vorgestellten Beschreibung von WFMS soll in diesem Abschnitt das Einsatzspektrum heutiger WFMS aufgezeigt werden. Wir werden diese Charakterisierung workflow-geeigneter Prozesse in Kapitel 4 aufnehmen und zeigen, dass wissensintensive Prozesse eine Erweiterung des gegenwärtigen Einsatzspektrums von WFMS darstellen.

Die prozessorientierte Herangehensweise an Unternehmensorganisation bedeutet immer auch eine Betrachtung möglichst effizienter Unterstützung der Prozesse durch Informations- und Kommunikationstechnologie. WFMS sind eine mögliche Variante dieser Unterstützung. Allerdings eignet sich Workflow-Management-Technologie nicht zur Unterstützung eines jeden Prozesses. Der Erfolg von Workflow-Management-Projekten hängt damit ganz wesentlich von der Frage ab, ob WFMS überhaupt die geeignete Unterstützung für den betrachteten Prozess bieten können. Diese Frage führt neben einer generellen Betrachtung der Automatisierbarkeit von Prozessen, wie sie in der Betriebswirtschaftslehre angestellt wird, letztlich zu der Notwendigkeit, die generellen Eigenschaften von WFMS zu beurteilen und mit den Anforderungen des jeweiligen Prozesses abzugleichen.

In der Literatur existieren verschiedene Klassifikationen von Prozessen und Aufgaben. Eine grundlegende Klassifikation von Picot & Reichwald [PR87] unterscheidet Aufgabentypen anhand der Merkmale

- Problemstellung/Komplexität,
- Informationsbedarf,
- Kooperationspartner und
- Lösungsweg.

Aus den möglichen Ausprägungen dieser vier Merkmale werden die drei Aufgabentypen Einzelfall, sachbezogener Fall und Routinefall abgeleitet:

Ein Einzelfall ist durch eine hohe Komplexität ausgezeichnet. Informationsbedarfe und Kooperationspartner können nicht (oder nur in sehr geringem Maße) festgelegt werden, und der Lösungsweg ist offen. Einzelfälle werden daher als nicht formalisierbar eingestuft.

Unter einem sachbezogenem Fall werden Aufgaben verstanden, die durch eine wechselnde Problemstruktur gekennzeichnet sind. Die Informationsbedarfe sind dabei nicht genau festlegbar. Die Kooperationspartner wechseln häufig und können ebenfalls nicht im Vorhinein genau festgelegt werden. Der Lösungsweg wird als geregelt bis offen eingestuft. Daher werden sachbezogene Fälle als teilweise formalisierbar charakterisiert.

Ein Routinefall ist durch eine gleichbleibende Problemstellung gekennzeichnet. Sowohl der Informationsbedarf als auch Kooperationspartner und Lösungsweg können weitgehend im Vorhinein bestimmt werden. Der Lösungsweg wird als „weitgehend programmierbar“ beschrieben. Daher werden Aufgaben dieses Typs als vollständig formalisierbar angesehen.

Weitere existierende Klassifikationen von Prozessen und Aufgaben [PR95 und KU94] basieren größtenteils auf den aufgeführten Merkmalen und sollen hier nicht detailliert betrachtet werden. Picot & Rohrbach [PR95] erweitern die obige Klassifikation auf Prozess- und Aufgabentypen und unterscheiden dabei einmalige Prozesse, Regelprozesse und Routineprozesse.

Aufbauend auf der diskutierten Klassifikation von Prozessen ist nun zu untersuchen, welche dieser Prozesstypen durch heutige WFMS geeignet unterstützt werden können. In [GKSW98] haben wir dazu einen in der Praxis evaluierten Kriterienkatalog zur Identifikation workflow-gerechter Prozesse vorgestellt. Der Kriterienkatalog spiegelt im Wesentlichen die Erfahrungen aus verschiedenen Workflow-Management-Projekten wider, d.h. der Aspekt der Praxisorientierung stand bei der Auswahl und Charakterisierung der Kriterien im Vordergrund. Allerdings erhebt der Kriterienkatalog keinen Anspruch auf Vollständigkeit. In spezifischen Projekten können weitere nicht erfasste Eigenschaften von Prozessen die Eignung von WFMS als Unterstützung beeinflussen. Dennoch konnten eine Reihe von Kriterien identifiziert werden, die in Übereinstimmung mit den entsprechenden Projektpartnern als besonders relevant für workflow-gerechte Prozesse erachtet wurden. Dabei wurde neben z.B. einem hohen Grad an Arbeitsteilung, der Möglichkeit der wirtschaftlicheren Gestaltung des Prozesses und der häufigen Ausführung insbesondere die ähnliche Ausführung von Geschäftsfällen als notwendige Eigenschaft eines zu unterstützenden Prozesses angesehen. Dies lässt darauf schließen, dass WFMS heutzutage in überwiegenderem Maße lediglich zur Unterstützung von Routineprozessen eingesetzt werden. Diese These steht im Einklang mit den Aussagen und der Produktausrichtung vieler WFMS-Hersteller, die das Einsatzpotential ihrer Produkte insbesondere auf Grund der erhöhten Wirtschaftlichkeit der unterstützten Prozesse im Bereich der Routineprozesse sehen.

Fragen der flexiblen Ausführung von Prozessen durch erweiterte WFMS sind in den letzten Jahren im wissenschaftlichen Bereich ausgiebig diskutiert worden. Dabei lag der Schwerpunkt zumeist auf Konzepten zur Ablaufflexibilität, also der Möglichkeit, während der Laufzeit in Bezug auf die durchzuführenden Aktivitäten und ihre Reihenfolge vom vorgegebenen Prozessmodell abweichen zu können oder das Prozessmodell unvollständig zu definieren und so dem Bearbeiter Spielräume bei der Durchführung offenzuhalten¹. Die entwickelten Konzepte sind allerdings nur zu einem geringen Teil in kommerziellen Systemen umgesetzt worden, so dass lediglich wenige marktrelevante Systeme in der Lage sind, Prozesse zu unterstützen, die gemäß obiger Klassifikation zwischen Routineprozessen und Regelprozessen angesiedelt sind.

Die Möglichkeiten heutiger kommerzieller WFMS zur Unterstützung prozessbezogener Informationen wurden bereits im vorherigen Abschnitt aufgezeigt. In Bezug auf den Informationsaspekt sind sowohl datenorientierte als auch dokumentenorientierte WFMS zur Unterstützung von Routineprozessen gut geeignet. Durch die Möglichkeit, Prozessmodellldokumente explizit angeben zu können, können die weitgehend bestimmbaren Informationsbedarfe mittels Workflow-Management-Technologie abgebildet werden. Die Unterstützung weiterer Informationsbedarfe über die Prozessmodellldokumente hinaus ist für Routineprozesse nur in sehr geringem Maße erforderlich. Wir werden auf der Basis dieser Überlegungen in Kapitel 4 die Klasse der wissensintensiven Prozesse einführen und zeigen, dass wissensintensive Prozesse grundsätzlich workflow-geeignet sind, im Gegensatz zu Routineprozessen aber in Bezug auf den Informationsaspekt weitergehende Anforderungen an Workflow-Management-Technologie stellen.

2.2 Wissensmanagement

Die Frage, wie ein systematisches Management der Ressource Wissen erfolgen kann, wird sowohl in der wissenschaftlichen Welt als auch in Unternehmen in den letzten Jahren verstärkt diskutiert. Eine Vielzahl von Unternehmen hat, getrieben durch einen verstärkten Kosten- und Wettbewerbsdruck, entweder bereits Wissensmanagementprojekte durchgeführt oder plant, in der Zukunft Maßnahmen für ein systematisches Wissensmanagement zu ergreifen [BWPW98, ORG00]. In diesem Abschnitt sollen daher sowohl die für diese Arbeit relevanten Grundlagen des Wissensmanagements als auch spezifische Aspekte einer informationstechnischen Unterstützung erläutert werden. Diese dienen zusammen mit den in Abschnitt 2.1 dargestellten Eigenschaften von WFMS als Grundlage sowohl für die Diskussion der im folgenden Kapitel vorgestellten existierenden Arbeiten zur Integration von Workflow Management und Wissensmanagement als auch für den in dieser Arbeit entwickelten Ansatz zur Unterstützung wissensintensiver Prozesse durch erweiterte WFMS.

Wir geben zunächst in Abschnitt 2.2.1 einen Überblick über wesentliche Grundlagen des Wissensmanagements und stellen einige für diese Arbeit relevante Begriffe vor. Um eine Operationalisierung der strategischen Aufgabe Wissensmanagement zu unterstützen, existiert eine Vielzahl von Prozessmodellen oder Wissensmanagement-Frameworks, die die einzelnen Aktivitäten für ein systematisches Management von Wissen beschreiben. Wir orientieren uns in dieser Arbeit an den Bausteinen des Wissensmanagements von Probst et al. [PRR98], die in Abschnitt 2.2.2 erläutert werden. Erfolgreiches Wissensmanagement in Unternehmen erfordert Bemühungen, die die drei Aspekte Organisation, Menschen und Technologie gleichermaßen berücksichtigt und in eine entsprechende Unternehmenskultur integriert. Für die in dieser Arbeit relevanten Fragestellungen sind informationstechnische Lösungen für das Wissensma-

-
1. Die Literatur zum Thema der flexiblen Ausführung von Prozessen durch WFMS ist sehr umfangreich und soll hier nicht vollständig aufgearbeitet wiedergegeben werden. Die folgenden Arbeiten geben einen Einblick in die Thematik und eignen sich als Einstieg für weitergehende Studien: [SGJR96 und insbesondere KDB98] geben einen Überblick über die Thematik. Eine Auswahl wichtiger Arbeiten zum Thema der flexiblen Änderung von Prozessen bzw. der Evolution von Prozessmodellen sind [AD98, BK95, CCP96, EKR95, Herr95 und HHJN99]. Weitere Arbeiten [DHMM96, GPP96, HLSS99] zeigen Ansätze im Bereich deklarativer Prozessmodelle auf.

nagement von besonderer Bedeutung. Daher stellen wir in Abschnitt 2.2.3 das grundlegende Konzept des Organizational Memory Information Systems (OMIS) vor. Wir werden später zeigen, dass eine Unterstützung wissensintensiver Prozesse durch WFMS nur gelingen kann, wenn WFMS um geeignete Funktionalitäten eines OMIS erweitert werden. Schließlich wenden wir uns in Abschnitt 2.2.4 dem derzeit stark diskutierten Begriff der Wissensarbeit zu. Die vorgestellte Charakterisierung von Wissensarbeit dient zusammen mit dem in Abschnitt 2.1.4 diskutierten derzeitigen Einsatzspektrum von WFMS als Grundlage für die Positionierung der in Kapitel 4 eingeführten Klasse der wissensintensiven Prozesse.

2.2.1 Überblick und Begriffsbestimmung

In der Literatur existieren eine Vielzahl von Definitionen für den Begriff des Wissensmanagements. Zwei grundlegende Definitionen, die sowohl den systematischen Ansatz als auch den betriebswirtschaftlichen Aspekt der Wertschöpfung durch Wissensmanagement betonen, sollen im Folgenden exemplarisch aufgeführt werden. So definiert Wiig Wissensmanagement als „the systematic and explicit management of knowledge-related policies, practices, programs, and activities within the enterprise. The overall purpose of KM [knowledge management] is to maximize the enterprise’s knowledge-related effectiveness and returns from its knowledge assets and to renew them constantly to secure viability and profitability.“ [Wiig96]. In eine ähnliche Richtung geht die Charakterisierung von Davenport & Prusak: „Knowledge Management is a formal, structured initiative to improve the creation, distribution, or use of knowledge in an organization. It is a formal process of turning corporate knowledge into corporate value.“ [DP98].

Alternativ erfolgt eine Charakterisierung von Wissensmanagement oftmals durch eine Darstellung der Aufgaben, die an ein systematisches Wissensmanagement gestellt werden. Exemplarisch sei im Folgenden eine entsprechende Beschreibung von Aufgaben nach Krogh & Venzin (zitiert nach [Fais96]) aufgeführt.

- Erschließen von Wissen (Erfahrungen, Best Practices) für alle, die dieses Wissen im Rahmen ihrer organisatorischen Rolle benötigen
- Verfügbarmachen von Wissen am Ort und zur Zeit der Entscheidung
- Erleichtern des effektiven und effizienten Entwickelns von neuem Wissen
- Sicherstellen, dass jeder in der Organisation weiß, wo Wissen verfügbar ist
- Umsetzen dieser Kompetenzen in neue Produkte und Dienstleistungen

Sowohl Definitionen für Wissensmanagement als auch die Beschreibung von Aufgaben des Wissensmanagements bieten eine eher abstrakte Sicht auf die Thematik, ohne dass konkrete Hinweise für eine Umsetzung und insbesondere für eine informationstechnische Unterstützung gegeben werden. Wir werden daher im Folgenden den Begriff des Wissensmanagements Schritt für Schritt konkretisieren und zunächst den dieser Arbeit zugrunde liegenden Wissensbegriff untersuchen.

Der Wissensbegriff wird in der Literatur, insbesondere in verschiedenen Disziplinen, wie Informatik oder Philosophie, unterschiedlich definiert. Wir orientieren uns in dieser Arbeit an der Wissenspyramide nach Aamodt & Nygard, in der die Begriffe Zeichen, Daten, Information, Wissen und Aktion unterschieden werden [AN95]

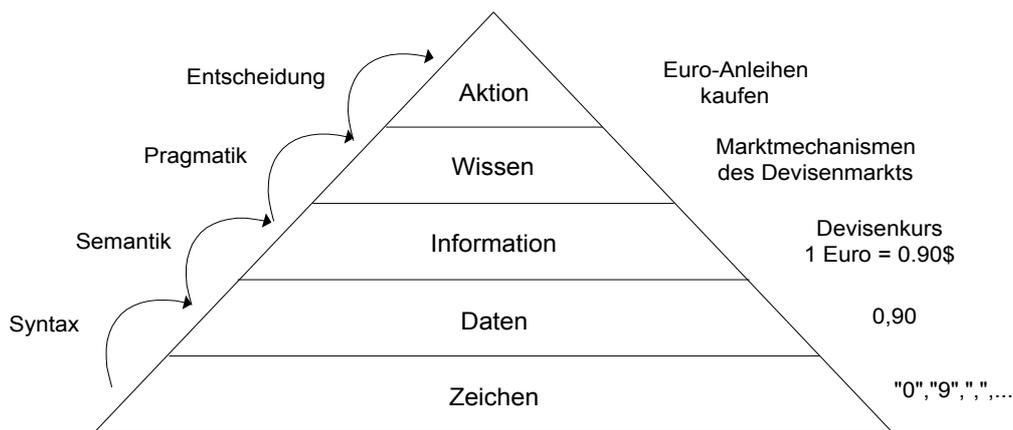


Abbildung 5: Wissenspyramide nach [AN95] (Beispiel an Anlehnung an [PRR98])

Ausgehend von einem Zeichenvorrat entstehen Daten durch die Anwendung von Syntaxregeln, also Regeln, die die Kombinierbarkeit von Zeichen vorgeben. Das Datum „0.90“ besitzt aber keinerlei Bedeutung, diese wird erst durch das Hinzufügen von Semantik erlangt. Die Daten werden dabei vom Empfänger in einem bestimmten Kontext interpretiert, und es wird eine Beziehung zur realen Welt hergestellt (im Beispiel versteht der Empfänger, dass es sich bei dem Datum um einen Devisenkurs des Euros im Vergleich zum US-Dollar handelt). Die Vernetzung von Informationen und die Nutzung der Informationen in einem bestimmten Handlungsumfeld (Pragmatik) führt nun zu Wissen. Dieses Wissen ist zweckgebunden und die Grundlage für Entscheidungen und Aktionen des Empfängers. Im angegebenen Beispiel kommt der Empfänger zu dem Schluss, dass aufgrund seiner Erfahrungen und weiterer Informationen ein Euro-Kurs von 0.90\$ als sehr günstig angesehen werden kann, und er kauft Euro-Anleihen bei seiner Bank.

Eine weitere Konkretisierung des Wissensbegriffs liefert die Unterscheidung von explizitem und implizitem Wissen nach Polanyi [Pola85]. Explizites Wissen ist beschreibbares und formalisierbares Wissen. Es ist nicht an einzelne Individuen gebunden und kann in verschiedenen Medien gespeichert sein. Es kann standardisiert, strukturiert und methodisch in sprachlicher und graphischer Form in Dokumenten, graphischen Symbolen, Diagrammen, Bildern und Datenbanken repräsentiert werden [BWP97]. Insbesondere kann explizites Wissen von IT-Systemen erfasst, gespeichert und verteilt werden.

Dagegen ist implizites Wissen subjektiv, stark kontextspezifisch und individuell an Personen gebunden. Es umfasst nach Nonaka & Takeuchi [NT97] sowohl technische Elemente, wie z.B. handwerkliches Geschick und Fertigkeiten, als auch kognitive Aspekte, also mentale Modelle, wie Paradigmen oder Überzeugungen, die Menschen helfen, ihre Welt wahrzunehmen und zu

definieren. Im Gegensatz zu explizitem Wissen lässt sich implizites Wissen überhaupt nicht oder nur unvollständig beschreiben oder formalisieren, indem es mittels Metaphern, Analogien oder Modellen begreifbar gemacht wird und damit in explizites Wissen umgewandelt wird.

Tabelle 1: Merkmale expliziten und impliziten Wissens

Explizites Wissen	Implizites Wissen
nicht an einzelne Personen gebunden	individuell an Personen gebunden
beschreibbar, formalisierbar	lässt sich höchstens unvollständig formalisieren
kann in verschiedenen Medien vorliegen und insbesondere auch von IT-Systemen verwaltet werden	kann bis zu bestimmtem Grad in Form von Metaphern, Modellen, Analogien in explizites Wissen umgewandelt werden

Nonaka & Takeuchi beschreiben in [NT97] einen Wissenslebenszyklus, nach dem explizites und implizites Wissen jeweils ineinander überführt werden kann (Abbildung 6). In der Phase der Sozialisation kann in z.B. Interessengemeinschaften oder anderen Gruppen bestehendes implizites Wissen erweitert und transferiert werden. Oftmals ist dafür nicht einmal ein Informationstransfer durch das Medium Sprache notwendig, wenn z.B. Lehrlinge handwerkliches Können von ihrem Meister durch Beobachtung und Nachahmung lernen. Durch Externalisierung von implizitem Wissen kann dieses in explizites Wissen umgewandelt werden. Dies geschieht z.B., wenn „best practices“ aus der Erfahrung heraus ausgewählt und in Form von Dokumenten abgelegt werden. Neues explizites Wissen durch die Verbindung bestehenden expliziten Wissens entsteht in der Phase der Externalisierung. Insbesondere die Vernetzung von in IT-Systemen abgelegten Wissensbeständen kann hier zu neuem explizitem Wissen führen. Der Prozess der Eingliederung explizites Wissens in das implizite Wissen durch Lernen findet schließlich in der Phase der Internalisierung statt.

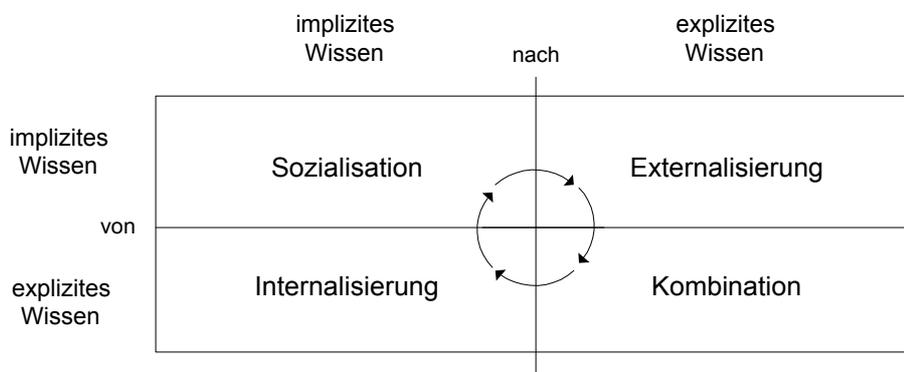


Abbildung 6: Wissenslebenszyklus [NT97]

Die begriffliche Trennung zwischen Information und Wissen bei der Beschreibung von informationstechnischen Lösungen für das Wissensmanagement wird oftmals unterschiedlich gehandhabt. Für diese Arbeit soll daher die folgende Sprachregelung getroffen werden: Wissen existiert streng genommen lediglich in den Köpfen von Menschen. Explizites Wissen, das in Form von papierbasierten oder elektronischen Dokumenten, Datenbankeinträgen, etc vorliegt, ist eigentlich im Sinne der Begriffsbildung von Aamodt & Nygard Information. Wir verwenden in dieser Arbeit dennoch den Begriff Wissen im Sinne von Nonaka & Takeuchi als Oberbegriff für sowohl implizites Wissen als auch explizites und dokumentiertes Wissen. In Bezug auf den Zusammenhang zwischen Wissen und Informationssystemen gilt das Folgende: Wissen kann zwar prinzipiell - oftmals allerdings nur unvollständig - in Informationssystemen repräsentiert werden, im Prozess des Speicherns wird es aber in rechnerbasierte Information umgewandelt, d.h. Informationssysteme versenden, präsentieren, bearbeiten, etc. nicht Wissen, sondern Informationen.

2.2.2 Bausteine des Wissensmanagements

Um ein systematisches Wissensmanagement in Unternehmen einführen zu können, sind zunächst die Gestaltungsdimensionen zu betrachten, die ein Wissensmanagementprojekt umfasst. In der Literatur (siehe z.B. [Albr93] oder überblickshaft in [WDA99]) werden diesbezüglich oftmals die „drei Säulen des Wissensmanagements“ beschrieben, nach denen ein erfolgreiches Wissensmanagement auf den Aspekten Organisation, Menschen und Technologie, eingebettet in eine entsprechende Unternehmenskultur, basiert (Abbildung 7). Die Organisation, d.h. sowohl die Aufbauorganisation als auch die Prozesse, ist dabei auf die konkreten Ziele des Wissensmanagements abzustimmen. Eine ideale Organisationsstruktur für Wissensmanagement existiert jedoch nicht: „Strukturen und Systeme sind immer ein Kompromiss zwischen sich widersprechenden Zielvorstellungen.“ [PRR98]. Eng mit dem Aspekt der Organisation hängt die zweite Säule des Wissensmanagements („Menschen“) zusammen. Nur durch die Bereitschaft des einzelnen, das Management von Wissen im Unternehmen als substantielles Ziel zu akzeptieren und durch die Weitergabe und Nutzung von Wissen zu fördern, können wissensorientierte Umgestaltungen der Organisation erfolgreich sein. Schließlich können informationstechnische Systeme in einer sehr starken Weise zum Gelingen von Wissensmanagementprojekten beitragen. Dabei muss allerdings stets beachtet werden, dass die Technologie nicht den entscheidenden Faktor in unternehmensweiten Wissensmanagementprojekten darstellt. Projekte, die sich zu stark auf technologische Lösungen konzentrieren, ohne die beiden anderen Säulen adäquat zu berücksichtigen, sind in den meisten Fällen zum Scheitern verurteilt.

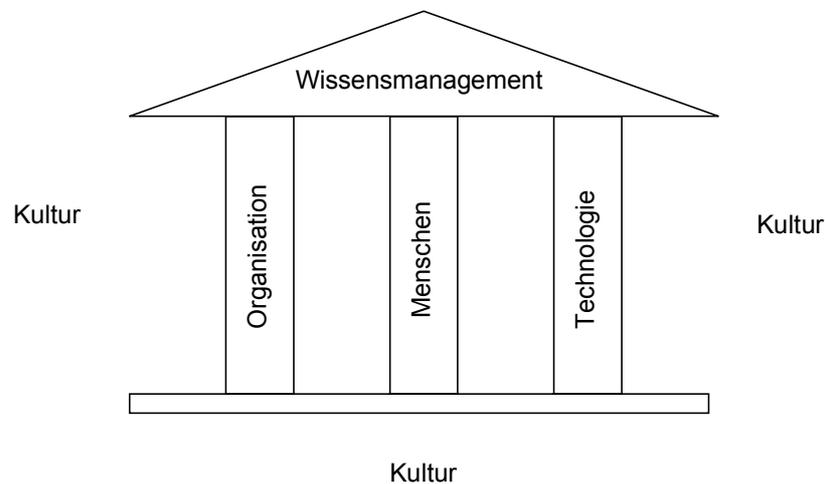


Abbildung 7: Drei Säulen des Wissensmanagements

Zur konkreten Planung von Wissensmanagementprojekten sind Aktivitäten und Prozesse zu identifizieren, mit denen die organisationalen Wissensbestände gelenkt und entwickelt werden können. In der Literatur finden sich eine Vielzahl von Modellen, die unterschiedliche Phasen oder Aktivitäten der Wissensverarbeitung unterscheiden und ggf. in eine zeitliche Reihenfolge setzen (Bsp. hierzu finden sich in [Allw98, NT97, RMGB98 oder Wiig98]). Holsapple & Joshi stellen in [HJ99] weitere Modelle oder „Frameworks“ des Wissensmanagement vergleichend vor und weisen darauf hin, dass jedes der untersuchten Modelle spezifische Aspekte des Wissensmanagements adressiert, keines aber jeweils alle anderen Modelle umfasst. Eine einheitliche Sicht auf die Aufgaben und Prozesse des Wissensmanagements, wie sie z.B. im Bereich des Workflow Managements existiert (Abschnitt 2.1.2), ist daher nicht gegeben.

Wir orientieren uns im Folgenden an einem Ansatz, der neben theoretischen Überlegungen auf der Analyse realer Problemstellungen beruht und in der Wissensmanagementliteratur breite Verwendung gefunden hat. Bei den von Probst et al. publizierten „Wissensbausteinen“ [PRR98] handelt es sich um eine Reihe von Kernaktivitäten oder Kernprozessen des Wissensmanagements, die alle mehr oder weniger stark miteinander verbunden sind (Abbildung 8). Wir werden uns in dieser Arbeit an den im Modell von Probst et al. vorgestellten Begriffen orientieren.

Die Anordnung der Bausteine folgt grundsätzlich dem Prinzip eines Regelkreises mit den Elementen Zielsetzung (Baustein „Wissensziele“), Umsetzung (innerer Kreislauf) und Messung (Baustein „Wissensbewertung“). Dieser Regelkreis setzt die Idee einer zielgerichteten Steuerung um, die sowohl strategische Aspekte als auch konkrete Zielsetzungen und deren Evaluation umfasst. Der innere Kreislauf enthält die Bausteine „Wissensidentifikation“, „Wissenserwerb“, „Wissensentwicklung“, „Wissens(ver)teilung“, „Wissensbewahrung“ und „Wissensnutzung“. Die einzelnen Bausteine wirken jeweils aufeinander ein. Daher sollten Maßnahmen des Wissensmanagements stets alle Bausteine berücksichtigen. Die einzelnen Bausteine werden im Folgenden kurz beschrieben.

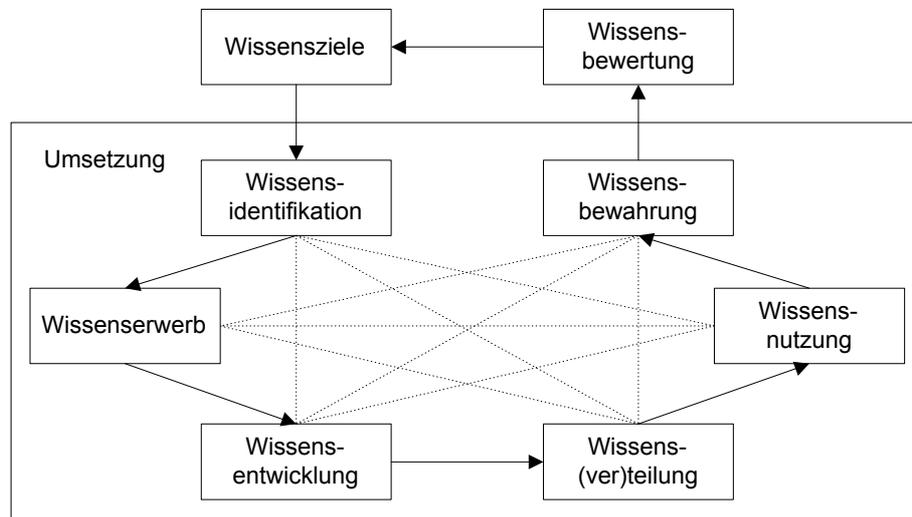


Abbildung 8: Bausteine des Wissensmanagements [PRR98]

Wissensziele legen die Richtung für die Maßnahmen des Wissensmanagements fest. Dabei werden normative Ziele (dienen der Schaffung einer wissensbewussten Unternehmenskultur), strategische Ziele (beschreiben zukünftigen Kompetenzbedarf des Unternehmens) und operative Ziele (sorgen für die Umsetzung des Wissensmanagements) unterschieden. In der Phase der Wissensbewertung erfolgt eine Evaluation des organisationalen Wissens in Bezug zu den definierten Zielen. Entsprechende Methoden zur Messung sind allerdings zur Zeit noch nicht etabliert.

Die Wissensidentifikation hat zum Ziel, das in der Organisation vorhandene Wissen strukturiert zu erheben und z.B. in Form von Wissenslandkarten oder Wissensstrukturdiagrammen darzustellen, um als Basis für Maßnahmen für eine weitere Verbesserung zu dienen. Der Wissenserwerb dient dazu, Fähigkeiten und Know-how aus Quellen außerhalb des Unternehmens einzukaufen. Dies kann einerseits in Form von explizitem Wissen, wie z.B. Dokumente oder Patente, oder in Form von Personalressourcen geschehen, indem also Experten rekrutiert oder Firmen akquiriert werden. Dagegen geht es bei der Wissensentwicklung um die Ausbildung neuer Fähigkeiten innerhalb der bestehenden Organisation. Diese Prozesse beruhen sowohl auf individueller Kreativität als auch auf kollektiven Lernprozessen, die durch eine entsprechende Unternehmenskultur ermöglicht werden müssen. Die (Ver)teilung (bzw. der Transfer) von Wissen bietet die Grundlage dafür, dass relevante Wissensbestände organisationsweit genutzt werden können. Hier steht die Frage im Vordergrund, welche Personen über welches Wissen verfügen müssen und wie die entsprechenden (Ver)teilungsprozesse effektiv und effizient unterstützt werden können. Nur wenn Wissen auch produktiv genutzt wird, stellt es einen Wert für die Organisation dar. Ein Beharren auf bewährten Vorgehensweisen kann die Wissensnutzung verhindern, selbst wenn die entsprechenden Wissensbestände geeignet verteilt wurden. Eine der größten Aufgaben bei der Einführung von Wissensmanagement in Unternehmen stellt schließlich die Wissensbewahrung dar. Eine fortwährende Selektion bewahrungs-

würdiger Wissensbestände, eine angemessene Speicherung und eine regelmäßige Aktualisierung des Wissens stellt sicher, dass Organisationen nicht einen Teil ihres Gedächtnisses verlieren bzw. dass stets aktuelles Wissen von den Mitarbeitern benutzt wird.

Um eine Unterstützung wissensintensiver Prozesse durch WFMS erreichen zu können, müssen bestimmte der vorgestellten Bausteine des Wissensmanagements geeignet in den Prozesslebenszyklus (Abschnitt 2.1.2) integriert werden. Wir werden diesen Aspekt in Abschnitt 5.5 im Rahmen der Diskussion von Anforderungen an zu erweiternde WFMS weiter detaillieren.

2.2.3 Organizational Memory Information Systems (OMIS)

Der Begriff „Gedächtnis“ beschreibt Fähigkeiten, die es ermöglichen, Wahrnehmungen über den Zeitraum der Wahrnehmung hinaus zu speichern und diese Wahrnehmungen zu einem späteren Zeitpunkt wiederum abzurufen [LMK98]. Auf der Ebene von Unternehmen wird der Begriff des „organisationalen Gedächtnisses“ (Organizational Memory) in dem Sinne benutzt, dass auch in einer Organisation Wissen in verschiedener Form gespeichert und zu einem späteren Zeitpunkt abgerufen wird. Dieses Wissen liegt einerseits in Form der Fähigkeiten und der Erfahrungen der Mitarbeiter vor, andererseits findet sich organisationales Wissen in expliziter Form in Dokumenten, Datenbanken, Dateien, etc. Damit bildet Organizational Memory die Grundlage für organisationales Lernen, also dem lernenden Anpassen der Aufbaustruktur und der Prozesse der Organisation [AS78]. Weitere in der Literatur oftmals synonym zu Organizational Memory verwendete Begriffe sind Corporate Memory, Collective Memory und organisationale Wissensbasis (einen umfassenden Überblick über Begriffe sowie die Theorie und Grundlagen des organisatorischen Gedächtnisses gibt [Lehn00]).

Da verschiedenste Disziplinen, wie z.B. Soziologie, Organisationslehre und Informatik, den Begriff Organizational Memory in unterschiedlicher Weise geprägt haben, existiert keine allgemeingültige Definition des Begriffs. In der Literatur zum Thema Wissensmanagement wird oftmals die Definition eines Organizational Memory von Walsh & Ungson verwendet. Demnach handelt es sich bei einem Organizational Memory um „stored information from an organization’s history that can be brought to bear on present decisions“ [WU91]. Stein & Zwass erweitern diese Sichtweise um das Attribut der Effektivität, wobei ein Organizational Memory aber nicht notwendigerweise zu einer höheren Effektivität der Organisation führen muss: „Therefore, we consider organizational memory to be the means by which knowledge from the past is brought to bear on present activities, thus resulting in higher or lower levels of organizational effectiveness“ [SZ95].

Eine informationstechnische Umsetzung des Konzepts des Organizational Memory stellen Organizational Memory Information Systems (OMIS) dar. OMIS bilden damit die zentrale Komponente für eine informationstechnische Unterstützung von Wissensmanagement in Unternehmen. Im Folgenden wollen wir den Begriff des OMIS näher charakterisieren und sowohl die Aufgaben eines OMIS als auch Schwierigkeiten der Umsetzung beleuchten. Dazu betrachten wir zunächst eine systemnahe Beschreibung von OMIS: „A Corporate Memory [OMIS] is an enterprise-internal application-independent information and assistant system. It stores large amounts of data, information and knowledge from different sources of an enterprise. These are represented in various forms, such as databases, documents, and formal knowledge-bases. The OMIS will be permanently extended to keep it up-to-date and can be

accessed enterprise-wide through an appropriate network infrastructure“ [KA98]. In Anlehnung an [KA98] werden die folgenden Aspekte als Kernanforderungen an ein OMIS identifiziert.

- **Sammlung und Strukturierung von Informationen aus unterschiedlichen Wissensquellen:**
Ein Großteil des Unternehmenswissens liegt in expliziter Form, beispielsweise in papierbasierten oder elektronischen Dokumenten, Datenbanken oder Groupware, vor. Eine zentrale Aufgabe eines OMIS besteht darin, diese unterschiedlichen Wissensbestände zu integrieren und zentral für den Benutzer zugänglich zu machen. Darüber hinaus sollte die Explizierung von Wissen der Bearbeiter in vernünftigem Umfang unterstützt werden. Dabei ist zu beachten, dass, wie oben bereits erläutert wurde, implizites Wissen nur unvollständig und teilweise nur unter erheblichem Aufwand dokumentiert werden kann.
- **Integration in die vorhandene Arbeitsumgebung:**
Um die Akzeptanz der Benutzer sicherzustellen, muss das OMIS in die vorhandene Arbeitsumgebung der Benutzer integriert sein. Das bedeutet, dass direkte Schnittstellen mit den Anwendungen existieren müssen, die die Mitarbeiter in ihrer täglichen Arbeit benutzen.
- **Aktive Präsentation relevanter Informationen:**
Ein OMIS sollte nicht nur eine passive Komponente darstellen, die auf Anfragen des Benutzers durch das Bereitstellen entsprechender Informationen reagiert. Durch die enormen Mengen an Wissen, die in Unternehmen anfallen, ist der einzelne Mitarbeiter überfordert, selbstständig nachzuforschen, in welchen Bereichen neues für ihn relevantes Wissen erzeugt wurde oder Wissen nicht mehr gültig ist. Daher sollte ein OMIS aktive Komponenten beinhalten, die dem Benutzer relevante Informationen automatisch bereitstellen bzw. den Benutzer über Veränderungen des im OMIS repräsentierten Wissensbestandes informieren. Der Benutzer sollte dabei stets die Kontrolle haben, über welche Wissensbestände er informiert werden möchte, um eine Informationsüberflutung durch das System zu vermeiden.
- **Sofortiger Nutzen für die Mitarbeiter:**
Beim Aufbau eines OMIS ist es von besonderer Wichtigkeit, dass der Nutzen des Systems für die Mitarbeiter sofort nach der Einführung sichtbar ist. Dies kann einerseits dadurch sichergestellt werden, dass existierende Wissensquellen so weit wie möglich eingebunden werden. Andererseits muss die Struktur der Wissensbestände im OMIS für die Benutzer verständlich sein. Nur so haben die Benutzer das Gefühl, dass der Aufbau und die Bewahrung des im OMIS repräsentierten Wissensbestandes direkt ihnen selbst und ihren unmittelbaren Kollegen, wie z.B. Mitgliedern ihrer Abteilung oder Projektmitarbeitern, zugute kommt, und sind daher motiviert, das System zu nutzen und Wissensbestände aktuell zu halten.

Die Abbildung des gesamten Wissens der Organisation in einem Informationssystem, wie es der Begriff Organizational Memory Information System vermuten lässt, ist illusorisch. Neben der eher technischen Problematik der Integration aller explizit im Unternehmen vorliegenden Wissensbestände ist insbesondere die Explizierung des gesamten impliziten Wissens, also aller Fähigkeiten der Mitarbeiter des Unternehmens, unmöglich. Hansen et al. unterscheiden diesbezüglich zwei Strategien für das Wissensmanagement in Unternehmen [HNT99]: Bei der Kodifizierungsstrategie wird das Wissensmanagement auf die Kodifizierung von Wissen in

Informationssystemen und die Integration vorhandener Informationssysteme fokussiert. Der Wissenstransfer wird über die gespeicherten Dokumente oder Daten durchgeführt. Demgegenüber konzentriert sich die Personalisierungsstrategie auf den interpersonellen Wissensaustausch. Informationssysteme spielen in diesem Fall im Vergleich zur Kodifizierungsstrategie eine eher untergeordnete Rolle. Sie haben nicht die Aufgabe, das explizierte Wissen zu speichern, sondern enthalten lediglich Verweise auf die entsprechenden Kompetenz- und Wissensträger. Diese Vorgehensweise wird in der Literatur auch als Skill Management [LW00] bezeichnet. Dabei ermöglicht die Entwicklung von sozialen Netzwerken, dass individuelles, implizites Wissen direkt unter den Beteiligten ausgetauscht werden kann. Bei der Konzeption eines OMIS müssen beide Strategien in Betracht gezogen werden. Insbesondere bei der Abbildung impliziten Wissens muss auf die Wissensexplizierung gemäß der Kodifizierungsstrategie in vielen Fällen sinnvollerweise verzichtet werden und stattdessen auf Komponenten zum Skill Management gemäß der Personalisierungsstrategie zurückgegriffen werden.

Die oben aufgeführten Anforderungen an ein OMIS werden zur Zeit noch von keinem kommerziellen oder prototypischen System vollständig umgesetzt. Daher ist die Abgrenzung von Systemen, die als OMIS gelten, schwierig und wird in der Literatur auch in unterschiedlicher Weise gehandhabt. Systeme, die in der Literatur als OMIS bezeichnet werden, unterscheiden sich in ihrer Funktionalität teilweise erheblich. Im Folgenden werden einige Systeme kurz vorgestellt, um die verschiedenen Funktionalitäten und Anwendungsgebiete von OMIS zu verdeutlichen (einen Überblick über weitere Systeme geben z.B. [ADK98, GFS98, Lehn00, LH98]).

Bei Answer Garden [AM90] handelt es sich um einen Forschungsprototypen, der ein Organizational Memory in Form eines kooperativen Hilfesystems realisiert. Der Benutzer kann einerseits in dem im System repräsentierten Wissensbestand navigieren und andererseits eine Frage an das System stellen, falls er weitergehende Informationen benötigt. Answer Garden leitet die Frage an einen zuständigen Experten weiter, der die Frage und die zugehörige Antwort in den Bestand des Systems einbringt, falls es sich um eine Frage von allgemeinem Interesse handelt. Bei Answer Garden handelt es sich um eines der ersten prototypisch implementierten OMIS. Das System wurde erfolgreich im Rahmen einer Feldstudie als Hilfesystem zu Problemen der Bedienung und Programmierung der X-Windows-Oberfläche eingesetzt [Acke94a]. Eine Weiterentwicklung des Systems, Answer Garden 2 [AM96], bietet zusätzliche Funktionalitäten, wie z.B. die Eskalation von Anfragen.

Die Intranet-Plattform Livelink [Open99] basiert prinzipiell auf Groupware-Funktionalitäten. Der Austausch von gespeicherten Informationen wird durch die Bereitstellung von Unternehmens-, Projekt- und persönlichen Arbeitsbereichen sowie durch die Verfügbarkeit von Dokumenten-Management und Suchfunktionen realisiert. Durch die Definition von Aufgaben, Aufgabenlisten und einfachen Workflows wird der Koordinationsaspekt unterstützt. Das Konzept der sogenannten „Channels“ bietet eine aktive Verteilung von Informationen an Benutzer, die diesen Channel abonniert haben.

Bei Fulcrum KnowledgeServer [Humm01] handelt es sich um ein integratives System. Fulcrum KnowledgeServer dient selbst nicht als Informationsspeicher, sondern ermöglicht die Integration verschiedener im Unternehmen bereits vorhandener Informationssysteme, wie z.B. Lotus Domino, das Intranet, das Dateisystem oder Dokumenten-Management-Systeme, die in einer

übergeordneten Ordnerstruktur als Wissenslandkarte dargestellt werden. Suchfunktionalitäten und aktive Agenten zur Informationsrecherche unterstützen das Auffinden von relevanten Informationen.

Auf Grund des heterogenen Charakters heutiger OMIS sind prinzipiell eine Menge von Basistechnologien zur Realisierung eines OMIS geeignet. Zu diesen Technologien zählen z.B. Data Mining, Dokumenten-Management-Systeme, Groupware, Filter- und Agentensysteme oder Intranet-Technologien, wobei in der Literatur oft noch weitere Technologien genannt werden (vgl. z.B. Habe99, LMK98, Warg98, WDA99 oder Wiig96). Die Diskussion dieser Fragestellung soll an dieser Stelle nicht vertieft werden, da die Basiskomponenten, die einen Beitrag für eine konkrete OMIS-Realisierung erbringen können, stets von den konkreten Anforderungen an das OMIS abhängen. Ein System wie Answer Garden stellt in Bezug auf die Realisierung vollkommen andere Anforderungen als z.B. DocsFulcrum, und daher sind allgemeine Betrachtungen, die die Obermenge aller möglichen Basistechnologien für konkrete OMIS aufzeigen, nur bedingt hilfreich.

Bei der Betrachtung des Einsatzes von OMIS fällt auf, dass die Systeme oftmals nicht zur Verwaltung des gesamten Wissens eines Unternehmens, sondern nur für thematisch abgegrenzte Wissensbestände oder eingeschränkte Benutzergruppen eingesetzt werden oder - wie bei Answer Garden der Fall - überhaupt nur für diesen begrenzten Einsatz konzipiert sind. Dies basiert auf der Erkenntnis, dass die Konzeption eines Systems, das sowohl die oben beschriebenen Anforderungen an ein OMIS erfüllt als auch das gesamte Wissen einer Organisation unterstützt, in den meisten Fällen illusorisch ist. Hier spielt insbesondere der problematische Umgang mit Kontext bei der Transformation von Wissen zwischen Autor und Leser durch ein OMIS eine entscheidende Rolle (vgl. [Acke94b und AH98] für eine vertiefende Diskussion dieses Aspekts). Informationen in einem OMIS, die insbesondere zwischen unterschiedlichen Organisationseinheiten und mit einer hohen zeitlichen Verzögerung zwischen Erstellung und Nutzung der Information ausgetauscht werden, müssen das richtige Ausmaß an Kontextinformationen beinhalten. Dabei muss einerseits die Information detailliert genug sein, dass sie dem Empfänger Nutzen bringt, andererseits benötigt der Empfänger nicht den vollen, für ihn ggf. verwirrenden Kontext der Erstellung der Information durch den Sender. Ackerman & Mandel propagieren daher ein "memory in the small" [AM95], das bei der Lösung lediglich bestimmter Aufgaben unterstützend wirkt. Diesem auch „task-based memory“ genannten Ansatz räumen die Autoren größere Erfolgchancen im Vergleich zu echten organisationsweit eingesetzten OMIS ein, da erstere direkter in die täglichen Aufgaben der Benutzer eingebettet sind und damit eine direktere Unterstützung bei der Bearbeitung von Aufgaben liefern können.

Als Fazit aus der dargestellten Betrachtung von OMIS sei noch einmal hervorgehoben, dass sich die Systeme, die sowohl in der wissenschaftlichen Literatur als auch von Systemherstellern als OMIS bezeichnet werden, erheblich in ihrem Funktionsumfang unterscheiden. „Echte“ OMIS, die die genannten Anforderungen erfüllen und wirklich das gesamte relevante Wissen einer Organisation unterstützen, sind zur Zeit noch Zukunftsmusik. Das in dieser Arbeit vorgestellte Konzept folgt daher dem eher pragmatischen Ansatz des „task-based memory“, indem ein OMIS für das Anwendungsfeld ausschließlich prozessbezogenen Wissens als Erweiterung eines WFMS konzipiert wird. Wir sprechen im Folgenden auch von einem prozessbezogenen OMIS¹.

Der Aspekt der Integration von Wissensmanagement (und hier insbesondere von OMIS) und der Gestaltung und Durchführung von Prozessen wurde in diesem Abschnitt nicht betrachtet. Wir werden auf der Basis der hier beschriebenen Grundlagen sowohl in Kapitel 3 verschiedene existierende Ansätze zu dieser Thematik vorstellen als auch später unseren eigenen Ansatz zur Unterstützung wissensintensiver Prozesse durch WFMS formulieren.

2.2.4 Wissensarbeit

In Diskussionen über die zukünftigen Aufgaben und Produkte von Unternehmen wird oftmals der Begriff der Wissensarbeit genannt, ohne dass eine exakte Definition und Abgrenzung des Begriffs erfolgt. So zitieren Probst et al. in [PRR98] eine Untersuchung, nach der in den USA bereits 60 Prozent aller Beschäftigten Wissensarbeit verrichten und vier von fünf Arbeitsplätzen aus sogenannten wissensintensiven Industrien stammen, ohne dass aber eine Beschreibung der spezifischen Eigenschaften von Wissensarbeit angegeben wird. Diese Beschreibung ist aber notwendig, um Wissensarbeit von anderen Arbeitsformen, wie z.B. Sachbearbeitertätigkeit, abzugrenzen und die Möglichkeiten einer IT-Unterstützung durch Workflow Management beurteilen zu können. Wir werden daher in diesem Abschnitt eine Charakterisierung von Wissensarbeit aufbauend auf der in diesem Zusammenhang relevanten Literatur geben. Diese Charakterisierung bietet zusammen mit dem in Kapitel 2.1.4 erläuterten Einsatzspektrum heutiger WFMS die Basis für die Positionierung der in Kapitel 4 eingeführten Klasse der wissensintensiven Prozesse im Spektrum zwischen Routineprozessen und Wissensarbeit.

Eine grundlegende Beschreibung von Wissensarbeit geht auf Kidd zurück, der in [Kidd94] aufbauend auf einer Feldstudie verschiedene Aspekte angibt, die Wissensarbeit von Sachbearbeitertätigkeit unterscheiden. Demnach ist Wissensarbeit durch folgende Merkmale charakterisiert:

- Vielfältige Ergebnisse:

Wissensarbeiter lösen Probleme und generieren Ergebnisse abhängig von individuellen Einflussfaktoren, d.h. sie sind im Vergleich zu Sachbearbeitern weit weniger an externe Regeln, Prozeduren oder Vorgaben gebunden. Auch steht bei der Wissensarbeit im Vergleich zur Sachbearbeitertätigkeit, die hauptsächlich die Nutzung vorhandener Informationen zum Inhalt hat, die Erzeugung neuer Informationen im Vordergrund.

- Unterschiedliche Arbeitsweisen:

Wissensarbeit zeichnet sich durch spontane Änderung des Verhaltens und der Arbeitsweise aus. Insbesondere werden oftmals spontan neue Informationen gesucht, bearbeitet und neue Informationen generiert. Dies geschieht in Abhängigkeit von dem sich oftmals ändernden Kontext, in den die aktuelle Aufgabe eingebettet ist. Dagegen zeichnet sich Sachbearbeitertätigkeit durch eine Kontinuität der Arbeitsweise und der generierten Ergebnisse aus.

-
1. Da OMIS im ursprünglichen Sinne das Ziel haben, das gesamte Wissen eines Unternehmens zu verwalten, handelt es sich streng genommen bei dem Begriff „prozessbezogenes OMIS“, also einer Einschränkung auf prozessbezogenes Wissen, um einen Widerspruch. Wir verwenden im Weiteren dennoch diesen Begriff, um deutlich zu machen, dass die geforderte Erweiterung von WFMS grundsätzlich auf Funktionalitäten eines OMIS basieren soll.

- Geringe Abhängigkeit von abgelegten Informationen:

Wissensarbeiter sind nur zu einem geringen Anteil von elektronisch oder papierbasiert abgelegten Informationen bei der Erledigung ihrer Aufgaben abhängig. In der von Kidd durchgeführten Studie wird berichtet, dass die Menge an abgelegten Informationen, die Wissensarbeiter für ihre Arbeit als relevant erachten, gering ist, und Wissensarbeiter nur selten auf diese Informationen zugreifen.

- Dynamische Kooperations- und Kommunikationsbeziehungen:

Es ergeben sich durch dynamische Teamstrukturen häufig neue Kooperations- und Kommunikationsbeziehungen, die ebenso schnell wieder verändert oder beendet werden. Mitarbeiter, die zum überwiegenden Teil Sachbearbeitertätigkeiten durchführen, sind dagegen zumeist in starre Organisationsstrukturen mit festen Verantwortlichkeiten und Spielräumen eingebunden.

Weitere Beschreibungen bzw. Definitionen von Davis et al., Davenport und Borghoff & Pareschi [BP98] sind weitgehend konform mit der von Kidd vorgeschlagenen Charakterisierung. Davis et al. definieren Wissensarbeit als "a set of activities using individual and external knowledge to produce outputs characterized by information content." [DCEN91]. Davenport et al. geben in [DJB96] eine detailliertere Beschreibung von Wissensarbeit: "Knowledge work's primary activity is the acquisition, creation, packaging, or application of knowledge. Characterized by variety and exception rather than routine, it is performed by professional or technical workers with a high level of skill and expertise". Wissensarbeit beinhaltet demnach z.B. Aktivitäten der Forschung, Produktentwicklung, Ausbildung, Beratung oder Management-Aktivitäten zur Strategieentwicklung und Planung.

Eine weitere Charakterisierung von Wissensarbeit, die ähnliche Eigenschaften wie bei Kidd herausstellt, erfolgt in der Literatur [CW97, Shum98] durch Rückgriff auf die Arbeiten von Rittel. Rittel vertritt den Standpunkt, dass der klassische Ansatz der Systemanalyse, zunächst ein Problem zu definieren und zu analysieren und daraufhin eine Lösung zu produzieren, für eine bestimmte Klasse von Problemen, nämlich den sogenannten "wicked problems", nicht zu brauchbaren Lösungen führt. Rittel & Webber nennen in [RW73] zehn Eigenschaften von „wicked problems“. Die hier relevanten Aspekte werden im Folgenden zusammengefasst:

- Es gibt keine definitive Formulierung des Problems. Die Informationen, die benötigt werden, um das Problem zu formulieren, sind abhängig von den Ideen, wie das Problem zu lösen ist. Die Formulierung des Problems ist das eigentliche Problem.
- Es existiert kein Kriterium, das exakt angibt, ob das Problem gelöst ist. Daher sind Problemlösungen auch nicht als korrekt oder inkorrekt, sondern eher in Begriffen, wie „besser oder schlechter“ oder „gut genug“ zu beurteilen.
- Jedes „wicked problem“ ist einzigartig. Es gibt keine im Vorhinein beschreibbare Menge von möglichen Lösungen, und es existiert keine Menge von zulässigen Operationen, die die Basis der Problemlösung bieten.

Typische Beispiele für „wicked problems“ sind die Ausarbeitung eines Mission Statements, das Design eines Produkts, Softwaredesign oder strategische Planung. Wie aus der jeweiligen Charakterisierung und den zugehörigen Beispielen deutlich wird, beschreiben die Ansätze von Kidd und Rittel also sehr verwandte Sachverhalte, die im Folgenden unter dem Begriff Wissensarbeit zusammengefasst werden können.

Wir haben in diesem Abschnitt eine Charakterisierung des in der Literatur oftmals unscharf und widersprüchlich verwendeten Begriffs der Wissensarbeit vorgenommen. Aus dieser Beschreibung wird deutlich, dass Wissensarbeit nicht sinnvollerweise durch Workflow Management unterstützt werden kann. Das Problem, also die Aufgabe oder der Prozess, kann nicht im Vorhinein beschrieben werden, spontane Änderungen des Vorgehens sind der Regelfall, und die Kooperations- und Kommunikationsstrukturen sind laufenden Änderungen unterworfen. Die der Wissensarbeit inhärente Flexibilität und Dynamik in Bezug sowohl auf die Verarbeitung und Erzeugung von Informationen als auch auf die Kooperations- und Kommunikationsstrukturen („There are no predetermined task sequences that, if executed, guarantee the desired outcome.“ [DJB96]) widerspricht damit dem Grundprinzip des Workflow Managements, das diese Aspekte zu einem hohen Anteil standardisieren will. Eine durchgängige Unterstützung von Prozessen, die als Wissensarbeit bezeichnet werden können, durch WFMS ist daher nicht sinnvoll, wobei allerdings stark strukturierte Teilprozesse, wie z.B. Freigabeverfahren innerhalb eines Produktentwicklungsprozesses, durchaus workflow-geeignet sein können. Eine Erweiterung des Einsatzspektrums von WFMS muss daher in klarer Abgrenzung von Aktivitäten, die in den Bereich der Wissensarbeit gehören, vorgenommen werden (Kapitel 4).

2.3 Dokumentenmodellierung

Bei der Konzeption eines OMIS ist insbesondere die Frage zu beantworten, in welcher Form die im OMIS zu speichernden Informationen repräsentiert werden sollen und welche Möglichkeiten sich demzufolge für die Suche nach Informationen ergeben. Wie in Abschnitt 2.1.1 erläutert, verwenden wir den Dokumentbegriff zur einheitlichen Beschreibung unterschiedlicher Informationstypen. Daher skizzieren wir im Folgenden Ergebnisse aus dem Bereich des Information Retrieval, die die Modellierung und Repräsentation von Dokumenten erlauben. Nach einer kurzen Einführung (für eine grundlegende Einführung in die Thematik des Information Retrieval sei auf die entsprechende Literatur verwiesen) erläutern wir ein sichtenorientiertes Modell zur Modellierung von Dokumenten. Wir werden dieses Modell bei der Diskussion von möglichen Ansätzen zur Strukturierung prozessbezogenen Wissens aufnehmen (Abschnitt 5.3). Unterschiedliche Möglichkeiten bei der Modellierung von Dokumenten bestehen bzgl. der Repräsentation der Semantik des Dokumenteninhalts. Wir werden hierzu einige Möglichkeiten unterschiedlicher Komplexität skizzieren. Diese dienen als Grundlage für die Darstellung existierender Ansätze zur Integration von Workflow Management und Wissensmanagement im folgenden Kapitel.

Der wesentliche Gegenstand des Information Retrieval kann kurz als „inhaltliche Suche in Texten“ zusammengefasst werden [Fuhr97]. Abbildung 9 zeigt ein Grundmodell des Information Retrieval: In der linken Seite der Abbildung ist der Eingabevorgang dargestellt, in dem Daten

(oder speziell Dokumente) analysiert und in gespeicherte Informationen überführt werden. Die rechte Seite der Abbildung zeigt den Prozess des Retrievals. Dabei wird die benötigte Information durch Transformationen auf den gespeicherten Informationsbeständen erzeugt.

Im Folgenden wenden wir uns speziell dokumenten-orientierten Aspekten des Information Retrievals zu und stellen ein Dokumentenmodell vor, das mehrere Sichten auf den Inhalt eines Dokuments beinhaltet [MRT91]. Dabei umfasst

- die Layout-Sicht die Darstellung des Dokuments auf einem zweidimensionalen Medium,
- die logische Sicht die einzelnen Bestandteile eines Dokuments auf Typebene, wie z.B. Kapitel, Abschnitt, Überschrift, etc., und
- die semantische Sicht bezieht sich auf die Bedeutung des Dokumenteninhalts.

Abecker et al. erweitern diesen Ansatz in [ABHK98] um eine weitere Sicht, nämlich um

- die Kontextsicht, die alle Attribute umfasst, die nicht direkt im Dokument vorhanden sind.

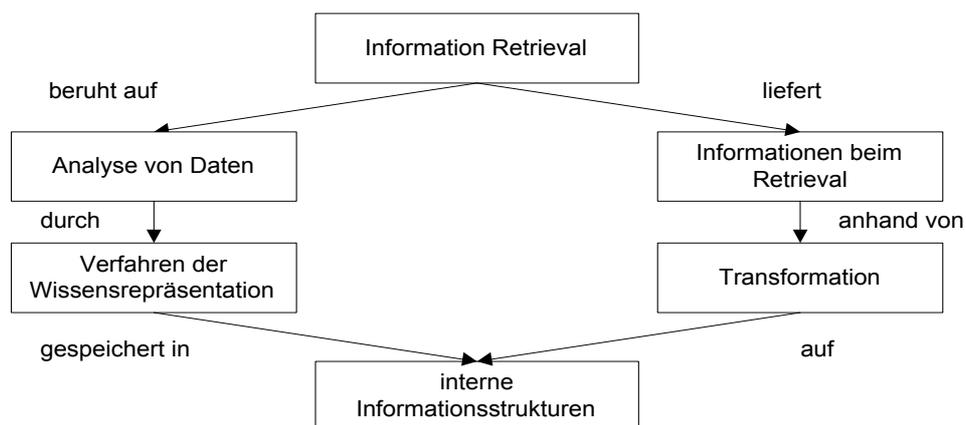


Abbildung 9: Grundmodell des Information Retrieval [Führ97]

Abbildung 10 fasst die einzelnen Sichten anhand eines Beispiels zusammen. Prinzipiell lässt sich das vorgestellte Modell nicht nur auf klassische Dokumente, sondern auch auf Objekte in Datenbanken anwenden. Dabei ist die semantische Sicht mit der logischen Sicht identisch, eine zusätzliche inhaltliche Repräsentation durch eine semantische Sicht ist bei Datenbankobjekten nicht notwendig.

Bei der Modellierung von Dokumenten, die Informationen unterschiedlichen Typs repräsentieren, sind die Kontextsicht und die semantische Sicht von besonderer Relevanz. Wir werden daher diese beiden Sichten im Folgenden ausführlicher beschreiben:

Kontextsicht

Abecker et al. verstehen unter der Kontextsicht die Menge der Attribute, die nicht direkt im Dokument selbst enthalten sind. Dazu unterscheiden sie zwei verschiedene Typen von Attributen [ABHK98]. Zum ersten Typ zählen Standardattribute, wie z.B. Autor, Erstellungsdatum oder Datum der letzten Änderung. Der zweite Typ setzt sich aus Informationen zusammen, die den Erstellungskontext des Dokuments beschreiben. Dieser Erstellungskontext wird durch Referenzen auf entsprechende modellierte Prozesse hergestellt, also indem z.B. spezifiziert wird, dass ein Dokument in einer bestimmten Aktivität eines Prozesses von einer Person mit einer bestimmten Rolle erzeugt wurde.

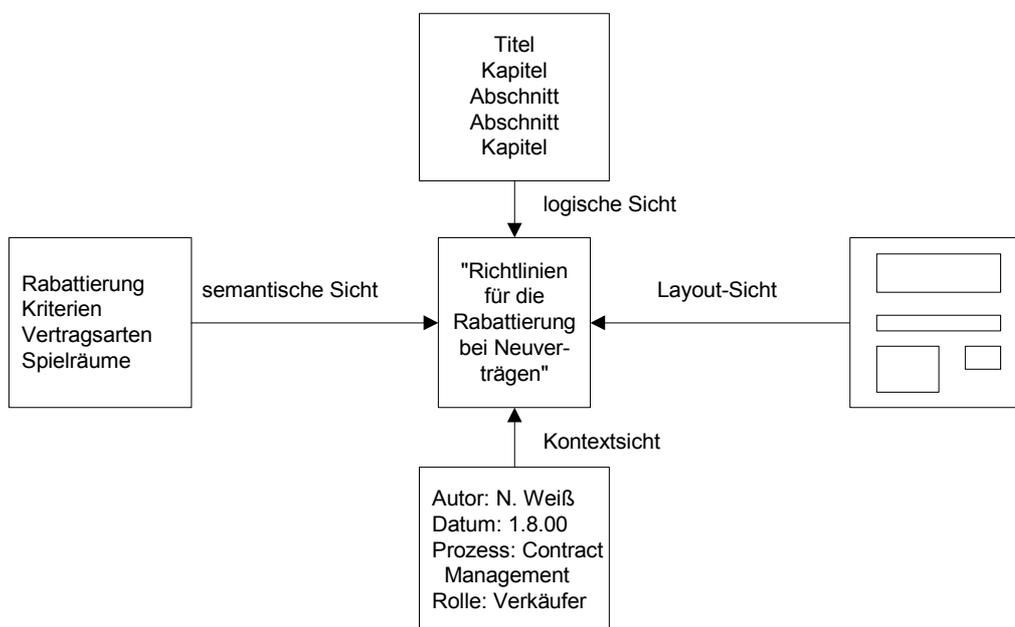


Abbildung 10: Sichten auf den Inhalt von Dokumenten (in Anlehnung an [Fuhr97])

Semantische Sicht

Um die inhaltliche Bedeutung von Dokumenten abzubilden, existiert eine Vielzahl von Ansätzen unterschiedlicher Komplexität. Dabei kann zunächst grundsätzlich unterschieden werden, ob eine zusätzliche inhaltliche Repräsentation des Dokumenteninhalts erstellt wird. Fuhr unterscheidet dazu die beiden Lösungsansätze „Freitextsuche“ und „semantischer Ansatz“ [Fuhr97].

Bei der Freitextsuche wird keine zusätzliche Repräsentation des Dokumenteninhalts erstellt, sondern es werden lediglich bestimmte Funktionen zur Suche im Text des Dokuments angeboten. Dabei wird der Text eines Dokuments reduziert, indem der Text zunächst in einzelne Worte zerlegt wird und durch Stoppworteliminierung nicht bedeutungstragende Worte ausgeschlossen werden. Die Suchfunktionalitäten beziehen sich dann auf die so reduzierte Folge

von Wörtern (vgl. „keyword-based content description“ in [ABHK98]). Die Freitextsuche stellt den klassischen Ansatz des Information Retrieval dar, der in vielen Systemen, wie z.B. Dokumenten-Management-Systemen, implementiert ist.

Das Ziel des semantischen Ansatzes ist die Erstellung einer zusätzlichen Repräsentation des Dokumenteninhalts, die von im Text verwendeten konkreten sprachlichen Formulierungen unabhängig ist. Dazu werden sogenannte Dokumentationssprachen verwendet, die ein einheitliches Vokabular festlegen und ggf. sogar darauf abzielen, die gesamte Semantik des Dokumenteninhalts zu formalisieren. Einen Ansatz für den Aufbau einer Dokumentationssprache bietet das Konzept der Ontologie. Ontologien bieten ein einheitliches Vokabular für einen bestimmten Anwendungsbereich und definieren die Bedeutung von Begriffen und die Beziehung der Begriffe untereinander (ein Überblick über Ontologien und ihre Anwendungsbereiche gibt [PB99]). Der Aufbau von Ontologien erfolgt im Allgemeinen manuell und kann - je nach Anwendungsgebiet - einen hohen Aufwand darstellen. Darüber hinaus existieren Formalismen aus dem Bereich der Künstlichen Intelligenz, wie z.B. semantische Netze, die eine möglichst weitgehende inhaltliche Repräsentation eines Dokuments zum Ziel haben. Offene Forschungsfragen betreffen die Kombination von Inferenzmechanismen und weitgehenden formalen Repräsentationsmöglichkeiten, um eine automatische Bestimmung von Dokumenten in Bezug auf Informationsbedarfe von Benutzern zu ermöglichen [KA98].

Beim Vergleich von Freitextsuche und semantischen Ansätzen liefern letztere oftmals bessere Trefferergebnisse durch die Abbildung verschiedener Textformulierungen auf eine einheitliche Bezeichnung und durch die Vermeidung von mehrdeutigen Begriffen. Allerdings wird dies durch einen höheren Aufwand bei der Eingabe neuer Dokumente in die Datenbasis erkaufte, da eine Klassifikation bzw. Indexierung von Dokumenten im Allgemeinen manuell erfolgt. Zusätzlich erfordert der Aufbau und die Pflege der Dokumentationssprache (z.B. bei Ontologien) oftmals einen hohen Aufwand.

3 Ansätze zur Integration von Workflow Management und Wissensmanagement

Ansätze für ein prozessbezogenes Wissensmanagement sind in den letzten Jahren in der wissenschaftlichen Literatur verstärkt diskutiert worden (siehe insbesondere [MAHM01, AMB01, AAAI00]). Das grundsätzliche Ziel eines prozessbezogenen Wissensmanagements liegt dabei in einer durchgängigen Unterstützung von Geschäftsprozessen (über die gesamten in Abbildung 2 dargestellten Phasen des Prozesslebenszyklus hinweg) mit Methoden und Techniken des Wissensmanagements. Eine Gesamtdarstellung der publizierten Ansätze zum prozessbezogenen Wissensmanagement ist für die in dieser Arbeit bearbeitete Fragestellung einer geeigneten Unterstützung wissensintensiver Prozesse durch WFMS allerdings zu umfangreich und zu unfokussiert. Daher werden wir in Abschnitt 3.1 eine einführende Darstellung verschiedener Aspekte prozessbezogenen Wissensmanagements geben und in den darauf folgenden Abschnitten 3.2 bis 3.9 prototypische Ansätze zur Integration von Wissensmanagement und Workflow Management erläutern. Schließlich gibt Abschnitt 3.10 eine Bewertung ausgewählter Aspekte der vorgestellten Systeme.

3.1 Ein Überblick über prozessbezogenes Wissensmanagement

Eine Vielzahl von Unternehmen hat in der Vergangenheit prozessorientierte Organisationsformen eingeführt. Daher liegt es nahe, dass bei der Einführung von Wissensmanagementsystemen die Geschäftsprozesse in besonderer Weise zu berücksichtigen sind, indem die durchgängige Unterstützung von Geschäftsprozessen durch Methoden des Wissensmanagements angestrebt wird. Wir stellen im Folgenden einige relevante Beiträge vor, die ein prozessbezogenes Wissensmanagement näher beleuchten.

Davenport et al. beschreiben in einem grundlegenden Artikel über die Unterstützung von Wissensprozessen [DJB96], dass die von Wissensarbeitern durchgeführten Aktivitäten und ggf. sogar die Zeit, Kosten und Qualität der Ergebnisse von Wissensarbeitern oftmals vom Management als "black box" angesehen werden. Daher fordern die Autoren eine prozessorientierte Herangehensweise, um diese Defizite der Unternehmensorganisation zu überwinden. Zu den Aspekten, die laut Davenport eine besondere Herausforderung für die Prozessorientierung von Wissensarbeitern darstellen, gehören geeignete unterstützende informationstechnische Systeme.

In eine ähnliche Richtung argumentieren Remus & Lehner [RL99]. Sie klassifizieren Wissensmanagementsysteme grundsätzlich als portal-orientierte, community-orientierte oder prozessorientierte Systeme, wobei Mischformen möglich sind. Für einen prozessorientierten Ansatz stellen die Autoren die folgenden Vorteile heraus:

- Orientierung an der Wertschöpfungskette:

Durch die Prozessorientierung kann insbesondere die geeignete Wissensbereitstellung für die Bearbeitung wertschöpfender und damit besonders unternehmensrelevanter Aktivitäten unterstützt werden. Die Prozesssicht dient damit als integrativer Faktor, um vorhandenes Wissen mit wertschöpfenden Aktivitäten zu verknüpfen.

- Kontextbereitstellung:

Prozesse bieten einen Teil des Kontexts, der für das Verständnis und die Internalisierung von Wissen notwendig ist. Der Kontext kann in Bezug auf die Prozessmodelle unterschiedlich weit gefasst sein: Bestimmtes Wissen bezieht sich auf eine einzelne Aktivität, andere Wissensbestände beziehen sich auf den gesamten Prozess oder auf eine einzelne Rolle, etc.

- Methodische Unterstützung der Durchführung von Wissensprozessen:

Im Bereich des Prozessmanagements existieren etablierte Methoden zur Modellierung, Auswertung und rechnergestützten Durchführung von Prozessen. Die Anwendung dieser Methoden auf das Wissensmanagement kann z.B. durch die Etablierung organisatorischer Rollen wie Prozessverantwortlicher, durch die Nutzung von Modellierungsansätzen oder durch die Verwendung von Methoden zur Ermittlung von Kosten und Nutzen erfolgen.

- Prozesse als Grundlage der Navigation:

Für das Design eines Wissensmanagementsystems bieten Prozesse eine für den Benutzer verständliche Möglichkeit zur Navigation in den verwalteten Wissensbeständen.

Eine ähnliche Argumentation für ein prozessbezogenes Wissensmanagement haben wir in [HGM01] vorgestellt.

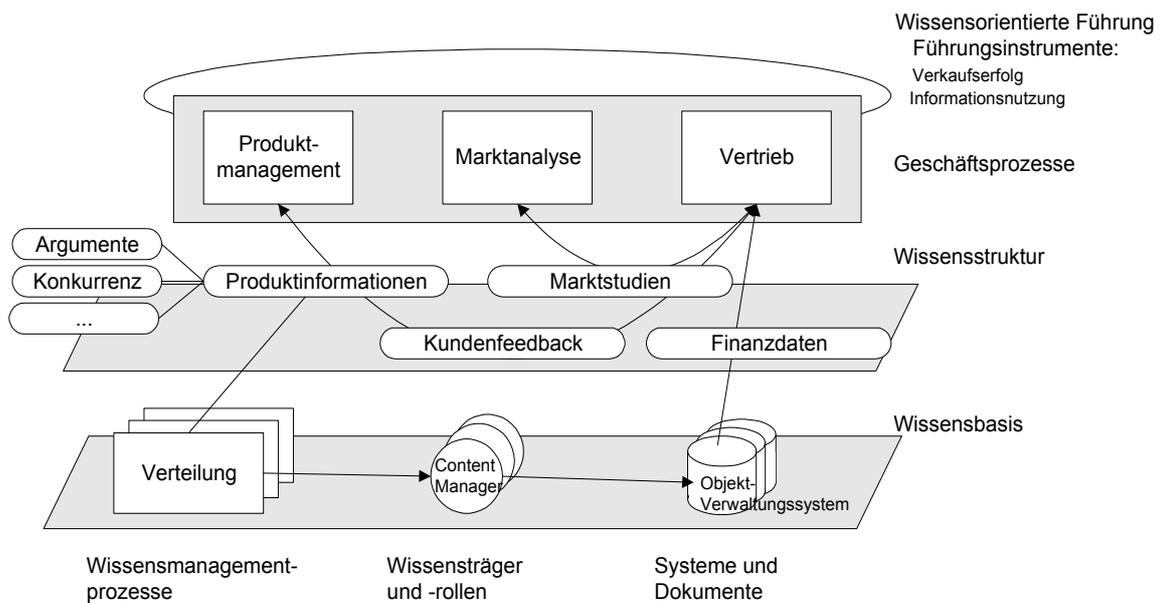


Abbildung 11: Modell des Business Knowledge Management, dargestellt anhand eines Beispiels [BVÖ99]

Um eine methodische Unterstützung von prozessbezogenem Wissensmanagement leisten zu können, sind insbesondere Modellierungssprachen und entsprechende Diagramme notwendig, die die wesentlichen Prozess- und Wissensaspekte integrieren. Als ein Beispiel soll hier der von Bach et al. in [BVÖ99] vorgestellte Ansatz des Business Knowledge Management dienen, der Instrumente der wissensorientierten Führung, die beteiligten Geschäftsprozesse, die Wissensstruktur und die Wissensbasis in einer Darstellung integriert (Abbildung 11). Das Modell stellt insbesondere die Wissensflüsse zwischen Geschäftsprozessen dar, beleuchtet die Rolle spezifischer Wissensmanagementprozesse (Verteilung) und gibt Hinweise auf die beteiligten Rollen und Informationssysteme. Die Methodik enthält darüber hinaus Darstellungen zur Verfeinerung der aufgeführten Aspekte.

Einen systematischen Ansatz zur prozessorientierten Planung, Analyse und Neugestaltung der Wissensverarbeitung schlägt Allweyer in [Allw98] vor. In Anlehnung an das Business Process Redesign (und damit, wie oben von Remus & Lehner vorgeschlagen, unter Ausnutzung etablierter Methoden aus dem Bereich des Prozessmanagements) wird ein Vorgehen zum „Knowledge Process Redesign“ vorgestellt. Unter Wissensprozessen (knowledge processes) werden dabei Prozesse verstanden, bei denen die Verarbeitung von Wissen eine hohe Bedeutung hat. Dabei kann es sich um operative (also „klassische“) Geschäftsprozesse (Bsp.: Kundenakquisition) oder um Prozesse handeln, deren Hauptzweck die Verarbeitung von Wissen ist (Bsp.: Aufbereitung von in Projekten angefallenem Wissen).

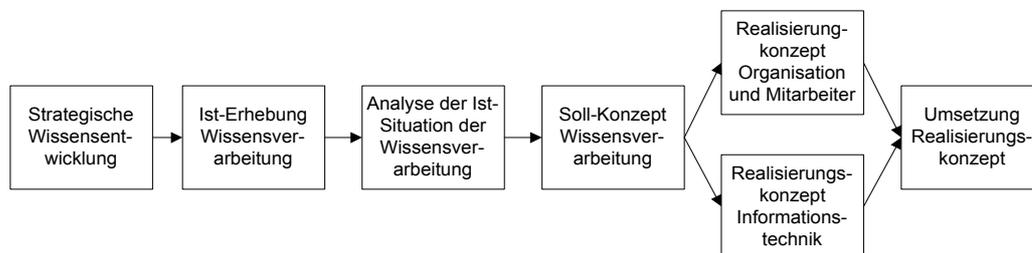


Abbildung 12: Vorgehen zum Knowledge Process Redesign [Allw98]

Die einzelnen Phasen des Vorgehensmodells sollen hier nicht im Detail beschrieben werden. Wichtig ist im Hinblick auf die Fragestellung eines geeigneten prozessbezogenen Wissensmanagements, dass in allen Phasen eine Integration von Wissen und Prozessen im Vordergrund steht: So werden z.B. bei der Wissensplanung wissensintensive Geschäftsprozesse identifiziert. Die Erhebung und Analyse der Ist-Situation erfolgt u.a. durch die Darstellung von Prozessen mit Hilfe von um wissensorientierte Aspekte erweiterten Modellierungssprachen, und bei der Soll-Konzeption werden bestehende Geschäftsprozesse verändert und neue Wissensprozesse definiert.

Die Fragestellung, inwieweit Workflow Management eine Unterstützung für das Wissensmanagement in Unternehmen liefern kann, wird von Pareschi in [Pare96] untersucht. Dabei orientiert sich der Autor an dem von Nonaka & Takeuchi vorgestellten Wissenslebenszyklus (Abbildung 6) und stellt für jede der vier Phasen Kombination, Sozialisation, Internalisierung und Externalisierung den möglichen Beitrag von Workflow Management dar: Kombination von Wissen im prozessbezogenen Sinn wird durch die kontinuierliche Prozessverbesserung

unterstützt, die durch die in WFMS gespeicherten Prozessmodelle vereinfacht wird. Die Sozialisation in Bezug auf prozessbezogenes Wissen erfordert sowohl Awareness-Mechanismen von WFMS, die den Bearbeiter auf angemessene Art mit allen notwendigen Informationen über den Status von Prozessen und Geschäftsfällen versorgen, als auch Möglichkeiten zur Kommunikation zwischen den Bearbeitern. Die Internalisierung von Wissen wird erleichtert, falls das Wissen strukturiert in Dokumenten oder Diagrammen oder verbal vorliegt. Die Historisierung von Geschäftsfällen durch das WFMS bietet hier die Grundlage für das Verständnis, wie und aus welchen Gründen Prozesse bearbeitet wurden, Entscheidungen getroffen wurden, etc. Schließlich kann Workflow Management die Externalisierung von Wissen fördern, indem bisher nicht dokumentierte Verfahrensweisen als Workflows dokumentiert und ausgeführt werden.

Wie eingangs bereits erläutert wurde, soll hier keine vollständige Übersicht über die publizierten Ansätze des prozessbezogenen Wissensmanagements gegeben werden. Es sollte vielmehr anhand der dargestellten Ansätze aufgezeigt werden, in welcher Art Methoden und Techniken des Prozessmanagements einen wichtigen Beitrag für das Wissensmanagement in Unternehmen liefern können. Im Folgenden werden wir darauf aufbauend der Fragestellung nachgehen, auf welche Weise WFMS durch wissensorientierte Methoden in Bezug auf Funktionalität und Anwendungsspektrum erweitert werden können bzw. in der Lage sind, wissensintensive Prozesse zu unterstützen. Dazu beschreiben wir die in dieser Hinsicht relevanten Forschungsansätze zur Integration von Workflow Management und Wissensmanagement. Bei der Beschreibung der Systeme lassen wir uns insbesondere von den folgenden Fragestellungen leiten:

- Welches Ziel wird mit einer wissensbezogenen Erweiterung des WFMS verfolgt?
Diese grundsätzliche Fragestellung bezieht sich darauf, welche Funktionalitäten eines WFMS durch wissensbezogene Methoden unterstützt werden sollen. Beispiele sind die kontinuierliche Verbesserung von Prozessen, die flexible Konfiguration und Ausführung von Geschäftsfällen oder die verbesserte Unterstützung der Durchführung einzelner Aktivitäten im Prozess.
- Welches Wissen wird verwaltet?
Dieser Aspekt beleuchtet den Inhalt (Wissen über Prozesse, Fachwissen, Erfahrungen, etc.) und die Struktur des Wissens (unstrukturierte Dokumente oder formal repräsentiertes Wissen), das durch das jeweilige System verwaltet wird.
- Worauf bezieht sich das verwaltete Wissen?
Diese Fragestellung bezieht sich auf den Kontext bzw. die Nutzungssituationen, für die Wissen verwaltet wird. Beispiele sind die Verwaltung von Wissen zu Prozessen, zu Geschäftsfällen und zu Aktivitäten und persönliche Arbeitsbereiche (um privates Wissen zu dokumentieren).
- Welche Operationen auf dem verwalteten Wissen stehen dem Bearbeiter zur Verfügung?
Hier ist insbesondere die Frage zu untersuchen, ob der Bearbeiter lediglich vorhandenes, ggf. zum Modellierungszeitpunkt beschriebenes Wissen nutzt oder ob er selbst Wissen, das er bei der Prozessbearbeitung gewonnen hat, in das System einbringen kann und damit zu einem kooperativen Wissensaufbau beitragen kann.

Die folgenden Systeme werden in den nächsten Abschnitten vorgestellt:

- WorkBrain (FORWISS Erlangen)
- Workware (SINTEF Oslo)
- Bramble (Universität Boulder, Colorado)
- KnowMore (DFKI Kaiserslautern)
- Ontobroker (Universität Kaiserslautern)
- EULE2 (Swiss Life)
- MILOS (Universität Kaiserslautern, Universität Calgary)
- PROMOTE (BOC GmbH, Universität Wien u.a.)

Es sei an dieser Stelle auf weitere Ansätze zur Integration von Workflow Management und Wissensmanagement verwiesen, die entweder laufende Forschungsarbeiten darstellen, deren Ergebnisse noch nicht beurteilt werden können [DHMS00], oder Ziele verfolgen, die nicht im Fokus unserer Betrachtung liegen: Hier sind die Arbeiten von Maus [Maus99] zur Integration von Funktionalitäten des Document Analysis and Understanding in WFMS und von Ruprecht et al. zur Individualisierung von Prozessmodellen [RPR99] auf der Basis von Kontextwissen zu nennen.

3.2 WorkBrain

Wargitsch et al. stellen in [WWT98] und [Warg98] ein flexibles WFMS (Flexware) vor, das eine OMIS-Komponente (WorkBrain) enthält. Flexware erlaubt die Möglichkeit, Workflows auf einem hohen Granularitätsniveau aus Bausteinen zusammenzusetzen, und bietet für jeden Baustein einen Katalog von Teilprozessen an. Zur Laufzeit kann der Bearbeiter einen Baustein verfeinern, indem er einen sinnvollen Teilprozess auswählt und diesen ggf. an seine Bedürfnisse anpasst. Dabei hat WorkBrain eine Doppelaufgabe zu erfüllen [Warg98]:

- Aus Sicht des Workflow Managements: Workflows mit Hilfe des im laufenden Betrieb gewonnenen Wissens zunehmend beherrschen und kontinuierlich verbessern.
- Aus Sicht des Wissensmanagements: Das in den Workflows erzeugte und enthaltene Wissen bewahren und für die Planung und Durchführung von Workflows nutzbar machen.

WorkBrain stellt dazu Mechanismen für ein „evolutionäres Workflow Management“ bereit. Dabei sollen zwei Evolutionszyklen unterstützt werden, die dem „double loop learning“ nach Argyris und Schön [AS78] entsprechen (Abbildung 13).

Der Evolutionszyklus 1 erstreckt sich über die Lebensdauer eines einzelnen Geschäftsfalls. Die Bearbeiter wählen aus der Workflow-Bibliothek, wie oben beschrieben, Bausteine aus und konfigurieren daraus eine Workflow-Instanz (learning by doing). Der Evolutionszyklus 2 entspricht einer reflexiven Betrachtung des ersten Evolutionszyklus. Aus den in WorkBrain protokollierten Workflow-Daten und Erfahrungen der Bearbeiter werden grundlegende Modifikationsbedarfe in Bezug auf die Modelle der Workflow-Bibliothek, die integrierten Anwendungen oder die Aufbauorganisation abgeleitet (learning by supervision).

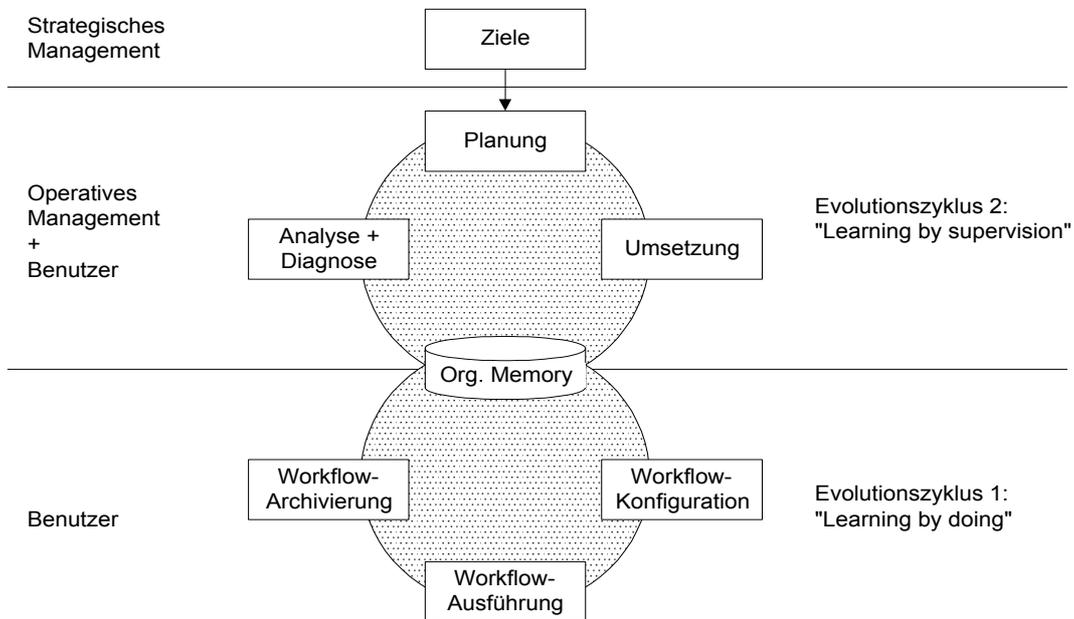


Abbildung 13: Doppelter Evolutionszyklus [Warg98]

WorkBrain bietet für die oben genannten Aufgaben eine Unterstützung, indem sowohl fallspezifisches als auch fallunabhängiges Wissen verwaltet wird (Abbildung 14 zeigt den logischen Aufbau von WorkBrain). Konkret unterstützt WorkBrain die folgenden OMIS-Funktionen bzw. Wissensbereiche:

- **Persönlicher Bereich:**

Über den persönlichen Bereich erhält der Bearbeiter Zugriff auf seine persönlichen E-Mails. Darüber hinaus erlaubt WorkBrain das Erstellen einer organisationsweit sichtbaren Homepage für jeden Mitarbeiter, die über die Erfahrungen und Qualifikationen jedes Mitarbeiters Aufschluss geben soll.

- **Organisation:**

Der Bereich „Organisation“ erlaubt den Zugriff auf die Organisationsdatenbank, wobei sämtliche Querverbindungen zwischen den Entitäten, wie z.B. Person, Rolle, Organisationseinheit, Stelle und Rechte, durch Hyperlinks abgebildet werden. Zusätzlich kann überprüft werden, welcher Bearbeiter welche Aktivität bereits wie oft durchgeführt hat, um z.B. eine Suche nach Experten zu erleichtern.

- **Workflows:**

Der Schwerpunkt der Arbeiten liegt auf dem Wissensbereich „Workflows“. Hier bietet WorkBrain den Zugriff auf die im WFMS gespeicherten Informationen zu z.B. Workflows, Aktivitäten und Geschäftsfällen und entsprechende Suchmöglichkeiten auf diesem Informationsbestand. Um eine flexible Konfiguration von Workflows zur Laufzeit zu unterstützen, bietet WorkBrain darüber hinaus eine auf Case Based Reasoning basierende Suche über

abgelaufene Geschäftsfälle an. Der Bearbeiter kann so über verschiedene Suchattribute, wie Workflow-Typ, beteiligte Bearbeiter oder Kunde, ähnliche Geschäftsfälle auffinden, die ihm die Konfiguration eines neuen Workflows vereinfachen.

- WIBIS (Workflow Issue Based Information System)

WIBIS ist eine Diskussionskomponente, die nicht nur das Ergebnis eines Lösungsprozesses, sondern ebenfalls die Lösungsfindung unterstützen soll. Dazu erlaubt WIBIS jedem Bearbeiter die Erzeugung neuer Diskussionsthemen zu technischen oder organisatorischen Problemen oder zum Thema „Workflow Management“ selbst.

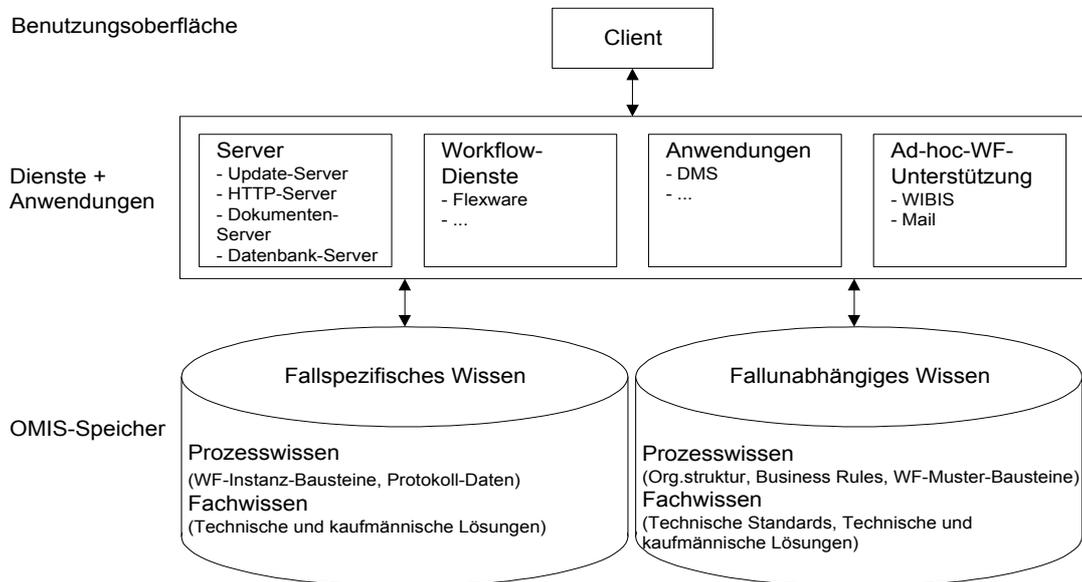


Abbildung 14: Logischer Aufbau von WorkBrain [Warg98]

Zusammenfassend handelt es sich bei WorkBrain um einen umfassenden Ansatz, der gemäß Abbildung 13 sowohl die Unterstützung von konkreten Geschäftsfällen - insbesondere unter dem Aspekt der flexiblen Konfiguration und Ausführung - als auch die kontinuierliche Verbesserung von Prozessen und der Organisation zum Ziel hat. Das verwaltete prozessbezogene Wissen bezieht sich sowohl auf die Informationen des WFMS als auch auf Wissen, das erst zur Laufzeit von den Bearbeitern gewonnen und dokumentiert wird (hauptsächlich durch WIBIS). Der Schwerpunkt liegt dabei auf der geeigneten Präsentation und Auswertung des ersten Aspekts, um mit Hilfe von Case Based Reasoning die flexible Konfiguration von Workflows zu unterstützen.

3.3 Workware

Das prototypische WFMS Workware zielt auf die Unterstützung von sogenannten „emergent workflows“ [JC99]. Der Begriff emergent workflow bezeichnet dabei einen teilweise strukturierten Prozess, der erst im Verlauf jeder einzelnen Durchführung, d.h. für jeden Geschäftsfall, von den Bearbeitern endgültig geplant wird. Die Modellierung und die Durchführung des Prozesses sind daher integriert zu betrachten. Dies bedingt die Möglichkeit, dynamische Änderungen an Prozessmodellen zur Laufzeit vornehmen zu können und unvollständige Prozessmodelle ausführen zu können. Um diese Anforderungen umsetzen zu können, werden die folgenden drei Aspekte für das System Workware fokussiert:

- Benutzerorientierte Prozessmodelle
- Integrierte Werkzeugunterstützung für die Planung, Koordination und Durchführung von Aufgaben
- Prozessbezogenes Wissensmanagement

Die ersten beiden Aspekte stehen für diese Arbeit nicht im Zentrum der Betrachtung. Daher sei für eine Übersicht über die entwickelte Prozessmodellierungssprache und die entsprechende Werkzeugunterstützung auf [CJ98] und [JC99] verwiesen.

Prozessbezogenes Wissensmanagement bezieht sich bei Workware grundsätzlich auf jede der einzelnen Ebenen der Prozessmodellierung: Bei der Planung und Neuplanung von Aufgaben durch das Erzeugen oder die Modifikation von Prozessmodellen gewinnen die Bearbeiter Wissen, das teilweise in den entsprechenden Prozessmodellen expliziert wird. Daher werden Prozessmodelle als Kernaspekt der Wissenserzeugung angesehen. Die Autoren beziehen sich auf den Wissenslebenszyklus nach Nonaka und Takeuchi (Abbildung 6) und führen aus, dass die Externalisierung von Wissen durch die im Workware-Ansatz entwickelten, leicht verständlichen Prozessmodelle unterstützt wird und die Kombination von Wissen durch verknüpfbare Prozessmodelltemplates ermöglicht wird. Ferner reflektiert die Möglichkeit zur Ausführung unvollständiger Prozessmodelle nach dem Prinzip der emergent workflows die Tatsache, dass Bearbeiter oftmals implizites Wissen besitzen, das nicht in Prozessmodellen dokumentiert werden kann. Zusammenfassend wird in [JC99] ein Referenzmodell für die Prozessmodellierung vorgestellt, das insbesondere den Aufgabenbereich des prozessbezogenen Wissensmanagements darstellt (Abbildung 15).

Das Referenzmodell enthält vier Stufen der prozessbezogenen Wissensrepräsentation, von allgemeiner Prozesslogik bis zu konkreter, situativer Durchführung von Aufgaben: Die Beschreibung der Prozesslogik meint eine noch relativ abstrakte Darstellung eines Prozesses und der Kernaktivitäten. Im nächsten Schritt wird durch die Spezifikation der Aktivitäten, Dekomposition und die Zuordnung von Ressourcen das Prozessmodell konkretisiert. Daraufhin wird das Prozessmodell auf eine einzelne Instanz, d.h. einen Geschäftsfall, konkretisiert, und es werden konkrete Ressourcen für den Geschäftsfall zugeordnet. Schließlich werden im letzten Schritt die modellierten Aufgaben tatsächlich durchgeführt.

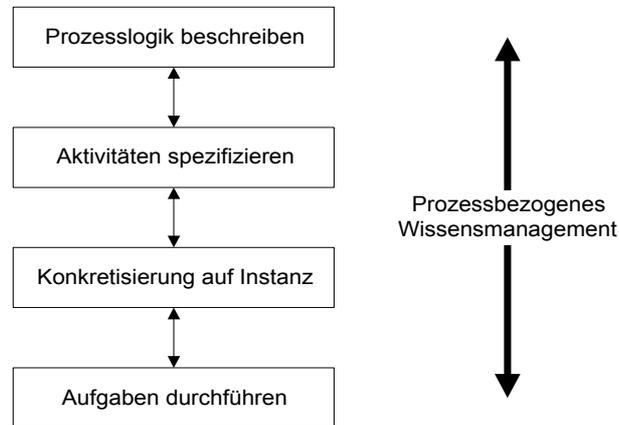


Abbildung 15: Referenzmodell für die Prozessmodellierung [JC99]

Prozessbezogenes Wissensmanagement findet auf allen Ebenen des Referenzmodells statt, wobei Workware insbesondere eine Unterstützung der unteren beiden Ebenen liefert. Dabei steht der Aspekt im Vordergrund, aus den Prozessmodellen abgelaufener Geschäftsfälle durch eine Evaluationsphase Wissen zu gewinnen, das zu einer Verbesserung der entsprechenden Prozessmodelltemplates führt. Dabei soll nicht nur das in den Prozessmodellen dokumentierte Wissen genutzt werden, sondern auch die Kommentare und Erfahrungen der Bearbeiter sollen als Grundlage der Evaluation der Prozessmodelle dienen. Die exakten Funktionalitäten, die Workware auf den einzelnen Ebenen zur Unterstützung von prozessbezogenem Wissensmanagement bietet, bleiben allerdings in der beschriebenen Literatur teilweise unklar.

Wie WorkBrain stellt auch Workware in Bezug auf prozessbezogenes Wissensmanagement die flexible Ausführung (im Sinne des Prinzips des emergent workflow) und die kontinuierliche Verbesserung von Prozessen in den Vordergrund. Dabei wird der Begriff des prozessbezogenen Wissens sehr stark auf das in Prozessmodellen explizierte Wissen eingegrenzt.

3.4 Bramble

Bei Bramble [BN95] handelt es sich um einen Forschungsprototypen, der eine verbesserte Unterstützung von unstrukturierten Aktivitäten auf der Basis von Workflow Management zum Ziel hat. Das System gründet auf einer zielbasierten Erweiterung von Information Control Networks (ICNs) [Elli79], wobei zusätzliche Informationen zur Unterstützung der Durchführung von unstrukturierten Aktivitäten auf formale Weise im Modell festgelegt werden können und durch entsprechende Werkzeuge vom Bearbeiter abgerufen werden können.

Bramble unterstützt lediglich einen bestimmten Anteil der Informationen, die Bearbeiter bei der Durchführung von unstrukturierten Aktivitäten benötigen (Abbildung 16). Zunächst werden „Task Information“ und „Instance Information“ unterschieden. Task Information besteht dabei aus den im Allgemeinen stets relevanten Regeln, Prozeduren oder Heuristiken, die von Bearbeitern zur Ausführung einer Aufgabe benutzt werden. Dazu wird Task Information in strukturierten (Workflow-)Modellen repräsentiert. Dagegen bezieht sich Instance Information

auf die zusätzlichen Informationen, die zur Durchführung einer Aktivität benötigt werden. Diese wird strukturiert in „External Information“, Informationen, die dem WFMS nicht bekannt sind (wie z.B. externe Dokumente), und „Contextual Information“, Informationen, die das WFMS verwaltet bzw. die aus dem WFMS-Informationsbestand abgeleitet werden können. Der im System Bramble realisierte Ansatz bezieht sich ausschließlich auf Contextual Information.

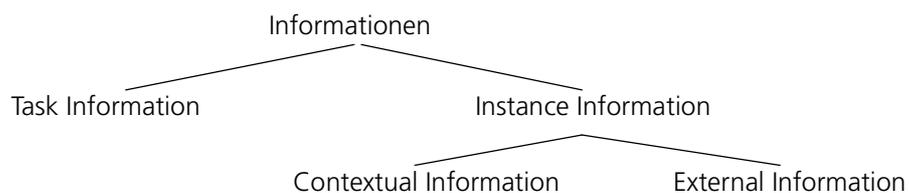


Abbildung 16: Relevante Informationen zur Durchführung unstrukturierter Aktivitäten

Der Modellierungsansatz der ICNs wird um die explizite Modellierung von unstrukturierten Aktivitäten erweitert, denen jeweils ein „Contextual Environment“ zugeordnet ist. Ein Contextual Environment setzt sich zusammen aus

- einer Menge von Zielen für die Aktivität, die als Zeichenketten angegeben werden,
- Kontextinformationen, die für jeden einzelnen Bearbeiter spezifizieren, welche zusätzlichen Informationen er für die Bearbeitung der Aktivität benötigt und
- eine Menge von Prozeduren, die angeben, wie die spezifizierte Kontextinformation dem Bearbeiter präsentiert wird.

Die Spezifikation der Kontextinformation erfolgt in einer eigenen Spezifikationssprache, die die Definition von Kontextinformationen auf unterschiedlichen Komplexitätsebenen erlaubt. So können z.B. auf der einfachsten Ebene Informationsbedarfe wie „Welche Aktivitäten folgen der aktuellen Aktivität?“ durch bestimmte Primitive ausgedrückt werden. Komplexere Funktionalitäten erlauben das dynamische Zusammenfügen von beliebigen dem WFMS bekannten Informationen zu neuen Ausdrücken, die dann in einer unstrukturierten Aktivität angezeigt werden können.

Mit Hilfe der beschriebenen Möglichkeiten können Bearbeiter jeweils unterschiedliche Informationen zur Bearbeitung unstrukturierter Aktivitäten heranziehen. Blumenthal und Nutt [BN95] nennen als Beispiel die Entscheidung über die Einstellung von Bewerbern auf eine wissenschaftliche Stelle. Während Professor A festlegt, dass er als Entscheidungshilfe für den Auswahlprozess die Information erhalten möchte, wieviele Bewerber von welchen Universitäten bereits akzeptiert wurden (um nicht zu viele Bewerber von derselben Universität zuzulassen), stützt sich Professor B stärker auf die Kompetenz seiner Kollegen, da er mit dem Fachgebiet des Bewerbers nicht exakt vertraut ist. Daher ist für ihn die Information relevant, inwieweit andere Professoren bereits den Bewerber eingeschätzt oder über ihn entscheiden haben. Die unterschiedlichen Informationsbedarfe können von den beiden Bearbeitern über die Spezifikation des Contextual Environments zu der entsprechenden Aktivität festgelegt werden.

Bramble ist ein Ansatz, der ein prozessbezogenes Wissensmanagement nur eingeschränkt unterstützt, da lediglich explizites Wissen und zwar nur die dem WFMS bekannten Informationen (Contextual Information) verwertet werden. Die Einflussnahme des Bearbeiters beschränkt sich darauf, die jeweiligen Informationsbedarfe zur Durchführung unstrukturierter Aktivitäten zu spezifizieren und diese zusätzlichen Informationen dann während der Durchführung der Aktivitäten zu nutzen. Auch wenn der Wissensbegriff nicht explizit genannt wird, liefert der beschriebene Ansatz Erkenntnisse über den Aspekt zusätzlicher Informationen zur Prozessdurchführung auf der Ebene konkreter Aktivitäten.

3.5 KnowMore

KnowMore zielt auf die Unterstützung von wissensintensiven Aktivitäten („knowledge-intensive tasks“ [ABHK98]), Aktivitäten hoher Komplexität und Relevanz für den gesamten Prozess, die eine hohes Maß an Wissen und Erfahrung des Bearbeiters benötigen. Um ein workflow-orientiertes Wissensmanagement zu ermöglichen, liegt der Kern des KnowMore-Ansatzes in einer geeigneten Repräsentation von wissensintensiven Aktivitäten, die einerseits in die Repräsentation des gesamten Prozesses integriert werden kann und andererseits spezifische Operationen zur Wissensbereitstellung und -nutzung für die Durchführung von wissensintensiven Aktivitäten unterstützt.

KnowMore stellt ein um OMIS-Funktionalitäten erweitertes WFMS dar und bietet sowohl dem Modellierer als auch dem Bearbeiter Unterstützung durch die folgenden Schritte [ABS99]:

- Zunächst wird der Prozess mit einem Prozessmodellierungswerkzeug und der Modellierungskomponente des WFMS modelliert.
- Für die wissensintensiven Aktivitäten wird das Modell jeweils um die Spezifikation von Anfragen an eine OMIS-Komponente (s.u.) erweitert, deren zugehörige Antworten für den Bearbeiter bei der Durchführung der wissensintensiven Aktivität relevant und hilfreich sind.
- Zur Laufzeit werden die Anfragen - parametrisiert mit den Attributen des aktuellen Geschäftsfalls - an das OMIS gerichtet, das dem Bearbeiter daraufhin Kontextinformationen zur jeweiligen wissensintensiven Aktivität bereitstellt.

Zur Modellierung wissensintensiver Aktivitäten sind also weitere Informationen notwendig, die über die Spezifikation von einfachen Aktivitäten, also klassischen Workflow-Aktivitäten, hinausgehen. Dazu wird zu einer wissensintensiven Aktivität eine Menge von Informationsbedürfnissen definiert. Die wichtigsten Attribute, die ein Informationsbedürfnis beschreiben, sind Vorbedingung, Agent-Spezifikation und Nachbearbeitung (eine exakte Beschreibung liefert [ABHK98]): Die Vorbedingung ist abhängig von im Prozessmodell definierten Workflow-Variablen oder weiteren Attributen des Prozesses und erlaubt, die Auswertung des Informationsbedürfnisses einzuschränken (Bsp.: Das Informationsbedürfnis muss nur ausgewertet werden, falls bestimmte Workflow-Variablen nicht belegt sind, also bestimmte Informationen noch nicht vorliegen). Der Agent ist ein externes Programm, das parametrisiert mit Workflow-Variablen oder internen Systemvariablen zur Laufzeit aktiviert wird, und dafür verantwortlich ist, die benötigten Informationen zu beschaffen. Unter bestimmten Bedingungen, die spezifiziert

werden können, ist eine Nachbearbeitung der Ergebnisse des Agenten notwendig. Mögliche Nachbearbeitungen sind z.B. die Berechnung aggregierter Werte aus den vorhandenen Informationen, die dem Bearbeiter angezeigt werden.

Die Durchführung einer wissensintensiven Aktivität liegt grundsätzlich in der Verantwortung des Bearbeiters. Der Workflow-Client zeigt dem Bearbeiter ein Editor-Fenster an, das eine textuelle Beschreibung der Aktivität liefert, die Eingangsvariablen anzeigt und dem Bearbeiter die Möglichkeit gibt, Werte von Ausgangsvariablen einzugeben. Parallel wertet das System die Spezifikation der Informationsbedürfnisse aus und bietet dem Bearbeiter an, falls die jeweilige Vorbedingung des Informationsbedürfnisses erfüllt ist, die entsprechenden Informationen anzuzeigen. Der Bearbeiter kann nun bestimmte Informationsbedürfnisse auswählen, woraufhin der entsprechende Agent aktiviert wird und ggf. nach einer Nachbearbeitung die Ergebnisse präsentiert werden. Legt der Bearbeiter im oben genannten Editorfenster Werte von Ausgangsvariablen fest, werden die ausgewählten Informationsbedürfnisse nochmals ausgewertet, und die Ergebnisse werden angezeigt. Abecker et al. beschreiben in [ABS99] beispielhaft den Prozess einer Bestellabwicklung, der als wissensintensive Aktivität, die manuell von einem Bearbeiter vorgenommen werden muss, die exakte Spezifikation des zu beschaffenden Artikels beinhaltet. Informationsbedürfnisse zu dieser Aktivität, die vom aktuell zu beschaffenden Artikel abhängig sind, umfassen z.B. technische Informationen über alternative Produkte, Hinweise auf in diesem Feld erfahrene Mitarbeiter oder interne Regeln über Entscheidungsspielräume bei der Beschaffung.

Abbildung 17 zeigt die logische Architektur des KnowMore-Ansatzes, der auf drei Schichten beruht. Auf der Anwendungsebene erlaubt der Ansatz die Spezifikation von wissensintensiven Aktivitäten wie oben beschrieben. Die Bereitstellung relevanter Informationen zur Durchführung wissensintensiver Aktivitäten wird durch eine Beschreibung der verfügbaren Informationen auf der Basis von Ontologien unterstützt. Der Inhalt eines Informationsobjekts wird durch Elemente der Domain Ontology repräsentiert, seine Struktur und der Zugriff auf die entsprechende Informationsquelle durch die Information Ontology festgelegt, und die Enterprise Ontology liefert Elemente zur Beschreibung des Kontexts eines Informationsobjekts, wie z.B. Prozessmodelle oder die Organisationsstruktur (eine detailliertere Beschreibung findet sich in [ABHK98]).

Die in [ABS99] beschriebene prototypische Implementierung des Systems umfasst eine eigenentwickelte Workflow-Engine, einen Workflow-Client, die verschiedenen Datenbanken für die oben beschriebenen Ontologien und für die Beschreibungen der konkreten Informationsobjekte und einen Transfermechanismus zur Einbindung von Prozessmodellen, die mit dem kommerziellen Prozessmodellierungswerkzeug ADONIS [BOC01] erstellt werden.

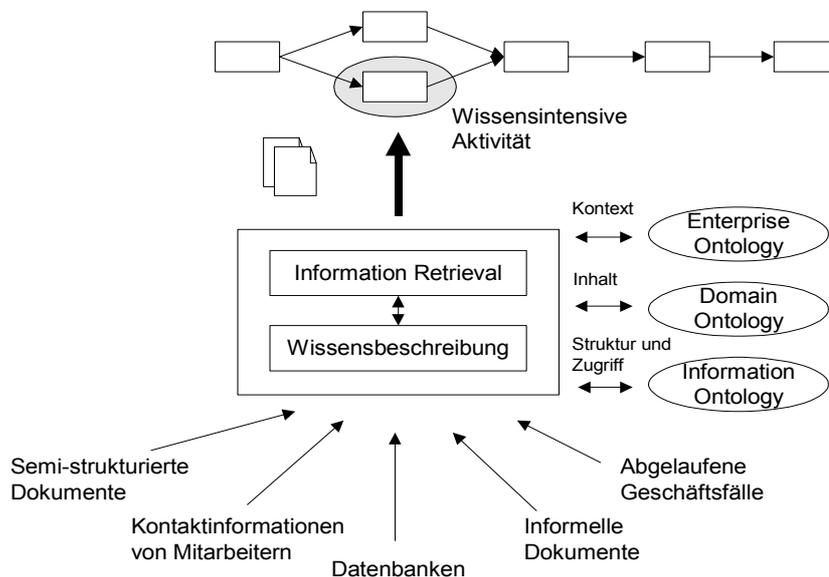


Abbildung 17: Logische Architektur von KnowMore [ABS99]

Wie bei Bramble handelt es sich bei KnowMore um einen Ansatz, der die Unterstützung wissensintensiver Workflow-Aktivitäten (und nicht die Prozess- bzw. Organisationsverbesserung insgesamt) zum Ziel hat. Die Einflussnahme des Bearbeiters zur Laufzeit beschränkt sich auf die Auswahl der für ihn relevanten Informationsbedürfnisse, die bereits zum Modellierungszeitpunkt spezifiziert worden sind. Anders als bei Bramble sind die Informationen, die das System dem Bearbeiter bereitstellt, nicht auf die vom WFMS verwalteten Informationen begrenzt. Vielmehr erlaubt KnowMore über den einheitlichen Beschreibungsmechanismus durch Ontologien die Integration verschiedener im Unternehmen vorhandener Informationsquellen, um dem Bearbeiter ein möglichst breites (im Sinne vieler eingebundener Informationsquellen), aber gleichzeitig auf sein aktuelles Informationsbedürfnis fokussiertes Spektrum an Informationen zu liefern.

3.6 Ontobroker

Staab & Schnurr beschreiben in [SS99, SS00] eine Umgebung auf der Basis des Systems Ontobroker, die darauf abzielt, die Prozessaspekte schwachstrukturierter Wissensarbeit zu integrieren und die Bearbeitung von Dokumenten in diesen Prozessen durch eine Bereitstellung der richtigen Wissensressourcen zum geeigneten Zeitpunkt zu unterstützen. Bei dem durch ein Wissensrepository repräsentierten Wissen handelt es sich um eine heterogene Menge von z.B. semi-strukturierten Dokumenten, WWW-Informationen oder Informationen in Datenbanken.

Die Basis dieses Ansatzes bildet das prototypische System Ontobroker, das - hier vereinfacht beschrieben - die folgenden Aspekte beinhaltet: Eine grundlegende Ontologie definiert Konzepte, Attribute, Beziehungen und Regeln für das Anwendungsgebiet. Dokumente werden mit Hilfe der Elemente der Ontologie annotiert und somit inhaltlich beschrieben. Über eine Komponente („Crawler“), die den Bestand der annotierten Dokumente nach Fakten durchsucht, gelangen diese Fakten in eine Wissensdatenbank. Die Wissensdatenbank wird schließlich von einer Inferenzmaschine durchsucht, um Anfragen des Benutzers zu bearbeiten und entsprechende Antworten für den anfragenden Benutzer zu erstellen.

Eine Unterstützung von prozessorientierten Aufgaben findet nach folgendem Ansatz statt: Bei Wissensarbeit handelt es sich zwar um unstrukturierte Prozesse in dem Sinn, dass nur auf sehr grob granularer Ebene konkrete Aktivitäten und deren Reihenfolge spezifiziert werden können, die notwendigen Dokumente zur Erfüllung der Aufgabe und deren Struktur kann allerdings auf der Modellebene festgelegt werden. Die Dokumente selbst werden als die Objekte angesehen, die - ggf. durch komplexe Aufgaben bearbeitet - den Fortschritt des Prozesses repräsentieren und deren Status jeweils die aktuellen Wissensbedarfe des Bearbeiters festlegt. Beispielhaft wird die Erstellung eines Projektplans beschrieben, der auf einer Prozessebene lediglich durch die Sequenz von drei Aktivitäten beschrieben werden kann (Abbildung 18), wobei das Erzeugen und schrittweise Ausfüllen des Dokuments, das den Projektplan repräsentiert, die eigentliche komplexe Aufgabe darstellt und abhängig vom Status des Dokuments unterschiedliche Wissensbedarfe zur Unterstützung der aktuellen Teilaufgabe bedingt. Der Bearbeiter besitzt also unterschiedliche Wissensbedarfe (Wissen über Kompetenzen von möglichen Projektmitgliedern, Wissen über Verfügbarkeit der Projektmitglieder, Wissen über freie Ressourcen, etc.), je nachdem welche Felder oder Sektionen des Dokuments bereits ausgefüllt sind bzw. sich aktuell in Bearbeitung befinden.

Die Prozesssicht wird durch SGML-Netze spezifiziert, wobei Aktivitäten lediglich auf einer sehr grob granularen Ebene beschrieben werden. Bei SGML-Netzen handelt es sich prinzipiell um Petri-Netze, die den Fluss von SGML-Dokumenten darzustellen erlauben und bei denen in Transitionen Operationen auf den SGML-Dokumenten ausgeführt werden. Die eigentliche Wissensarbeit geschieht allerdings jeweils innerhalb der Transitionen des Netzes (diese entsprechen den Aktivitäten des Prozesses) durch die Bearbeitung der zugehörigen Dokumente. Um dem Bearbeiter dabei Wissen gemäß seines aktuellen Bearbeitungskontexts anbieten zu können, kann zu einer Transition die Spezifikation kontextabhängiger Sichten auf den durch Ontobroker klassifizierten und verwalteten Dokumentenbestand erfolgen. Die Anfragen, die diesen Sichten entsprechen (Abbildung 18), können direkt aus den zu bearbeitenden Dokumenten heraus vom Bearbeiter ausgeführt werden und liefern dem Bearbeiter kontextabhängige Ergebnisse.

Ist beispielsweise zu einer Transition eine kontextabhängige Sicht spezifiziert, die die Menge der Bearbeiter mit Kompetenzen zu bestimmten softwaretechnischen Methoden identifiziert, und hat der Bearbeiter bereits im in der Transition zu bearbeitenden Dokument „Projektplan“ die notwendigen Kompetenzen mit „XML“ spezifiziert, dann wird ihm automatisch ein Menüeintrag mit der Anfrage „Finde Mitarbeiter, die XML-Kenntnisse haben“ durch das System angeboten.

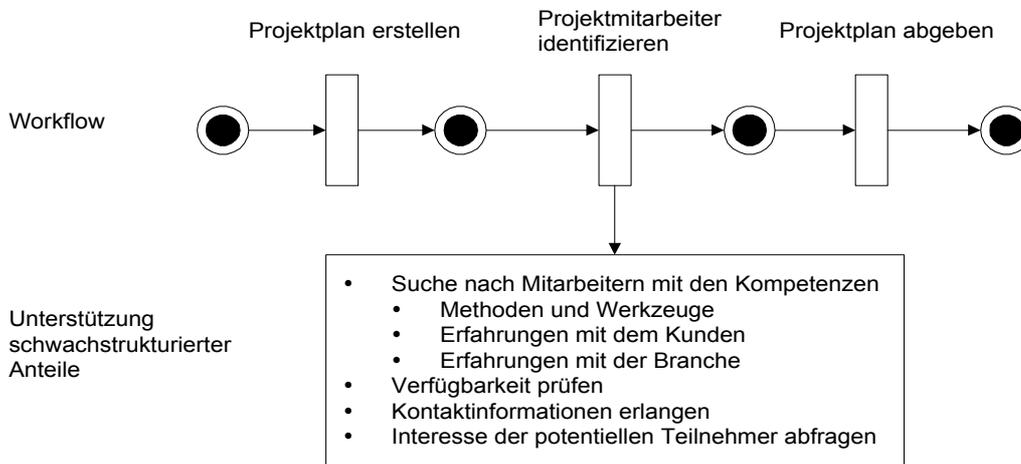


Abbildung 18: Integration von Workflow und kontextabhängigen Sichten [SS00]

Der Ontobroker-Ansatz ähnelt sowohl in Bezug auf die Zielsetzung (Bereitstellung von kontextabhängigen Informationen zur Bearbeitung von Workflow-Aktivitäten) als auch in Bezug auf die konzeptionelle Gestaltung (Ontologien als Grundlage zur Repräsentation von Dokumenten) sehr stark dem oben beschriebenen System KnowMore. Auch bei Ontobroker spielt der Bearbeiter eine passive Rolle, indem er zwar die vordefinierten Abfragen nutzen kann, um seine kontextabhängigen Wissensbedürfnisse zu erfüllen. Es ist aber nicht vorgesehen, dass der Bearbeiter integriert in die Prozessbearbeitung selbst prozessbezogenes Wissen expliziert, im System speichert und somit aktiv zu einem Wissenstransfer beiträgt.

3.7 EULE2

Bei EULE2 [Reim97] handelt es sich um ein wissensbasiertes System, das im Schweizer Versicherungskonzern Swiss Life Sachbearbeiter bei der Bearbeitung von Versicherungsvorgängen unterstützt. EULE2 bietet dem Bearbeiter eine aktive Hilfestellung und Führung bei der Ausführung von Aufgaben, deren Details dem Bearbeiter aufgrund der Komplexität der Aufgabe nicht exakt bekannt sind.

Eine Aufgabe kann als Graph modelliert werden, der ähnlich wie ein Prozessmodell eine Menge von Aktionen enthält, die in einer bestimmten Reihenfolge und durch bestimmte einschränkende Bedingungen auszuführen sind. Die Bedingungen für die Ausführung einzelner Aktionen ergeben sich dabei aus gesetzlichen Bestimmungen und unternehmensinternen Richtlinien. Im Gegensatz zu klassischen Prozessmodellen wird eine Aufgabe bei EULE2 stets von einer einzelnen Person durchgeführt. Das System unterstützt somit keine arbeitsteiligen Vorgänge.

Bei der Durchführung einer konkreten Aufgabe leitet EULE2 den Bearbeiter anhand des modellierten Aufgabengraphs. Gibt der Bearbeiter z.B. in einer Aktion bestimmte Daten ein, wählt EULE2 die entsprechenden zulässigen Folgeaktionen aus und bietet dem Bearbeiter

diese zur Auswahl an. Zusätzlich werden in jeder Aktion die benötigten Informationsobjekte aktiv bereitgestellt. Der Bearbeiter kann außerdem das System fragen, aus welchem Grund bestimmte Folgeaktionen nicht zulässig sind, woraufhin EULE2 die entsprechenden Informationen, z.B. Gesetzestexte, darstellt. EULE2 speichert das prozessbezogene Wissen, d.h. die Aufgaben, die Gesetze und Richtlinien und die verwendete Terminologie, in einer jeweils eigenen formalen Repräsentation, um eine automatische Auswertung der Wissensbestände und die aktive Führung des Bearbeiters durch das System zu ermöglichen.

Weiterentwicklungen des Systems [KJSR99] zielen auf die Integration von EULE2 als Wissensbasis in das WFMS Staffware. Während das WFMS die übergeordnete Steuerung eines arbeitsteiligen Prozesses übernimmt, leitet EULE2 den Bearbeiter jeweils bei der Ausführung einzelner Aktivitäten (wie oben beschrieben). Aufgaben in EULE2 können also als Subaktivitäten einer zugehörigen Workflow-Aktivität angesehen werden (Abbildung 19).

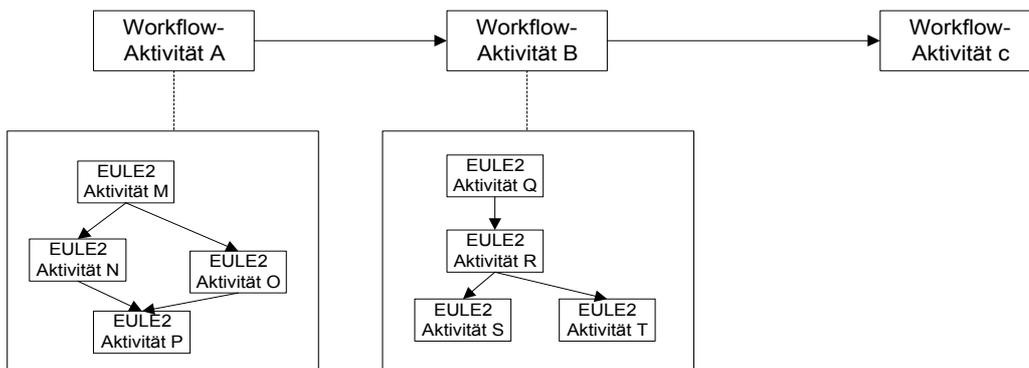


Abbildung 19: Integration von EULE2 in ein WFMS [KJSR99]

Bei EULE2 handelt es sich um ein wissensbasiertes System, das integriert in ein WFMS den Bearbeiter bei der Durchführung von einzelnen Aktivitäten unterstützt. EULE2 ist dabei sowohl aktiv, indem dem Bearbeiter die notwendigen Informationsobjekte bereitgestellt werden und zulässige Folgeaktionen zur Auswahl angeboten werden, als auch passiv, indem das System als Hilfesystem fungiert. Zu diesem Zweck basiert EULE2 auf einer formalen Repräsentation der einzelnen Aspekte des prozessbezogenen Wissens. Der Bearbeiter besitzt keine Möglichkeit, eigenes Wissen oder Erfahrungen bei der Bearbeitung von Aktivitäten in EULE2 einzufügen und somit zu einem Wissensaufbau beizutragen. Dies ist für den beschriebenen Anwendungsbereich angemessen, da EULE2 gesetzliche Vorgaben und unternehmensinterne Vorgaben verwaltet, für eine allgemeine Unterstützung wissensintensiver Prozesse allerdings nicht ausreichend.

3.8 Milos

Das Projekt MILOS (Minimally Invasive Long-term Organizational Support) [MD99, MH99] zielt auf die Unterstützung verteilter Softwareentwicklungsteams durch die Integration von Wissensmanagement, Projektplanung und Workflow Management. An eine entsprechende Workflow-Infrastruktur werden dazu die folgenden Anforderungen gestellt [MH99]:

- Bereitstellen von relevantem Wissen zu Workflow-Aktivitäten zum Zeitpunkt der Ausführung der jeweiligen Aktivität
- Integration von Wissensmanagement mit Projektplanung und Projektkoordination
- Unterstützung der kontinuierlichen Verbesserung der Organisation aufbauend auf der Evaluation der Prozessausführungen durch Methoden und Techniken des Wissensmanagements

Der MILOS-Ansatz unterscheidet dazu die folgenden drei Typen von prozessbezogenem Wissen:

- Generisches wiederverwendbares Wissen (generische Prozessmodelle):

Generische Prozessmodelle stellen wiederverwendbare Beschreibungen von Softwareentwicklungsprozessen dar. Primär enthalten sie die klassischen Prozessmodellaspekte, wie Aktivitäten, Ausführungsbedingungen, Ausführungsreihenfolgen von Aktivitäten, Informationsfluss, etc. Darüber hinaus können Prozessmodelle Kontextinformationen zu einzelnen Aktivitäten enthalten, die für den Bearbeiter relevant sind und Hintergrundwissen repräsentieren. Diese Kontextinformationen können entweder in Form von Dateien oder URL-Referenzen vorliegen oder als vordefinierte Abfragen im Prozessmodell hinterlegt werden, falls der exakte Wissensbedarf zum Modellierungszeitpunkt noch nicht bekannt ist. Eine Abfrage kann z.B. Informationen aus einer Erfahrungsdatenbank zu früheren Projekten abfragen oder an ein Dokumenten-Management-System gesandt werden.

- Wissen über einzelne Projekte (Projektpläne):

Projektpläne konkretisieren generische Prozessmodelle gemäß der Anforderungen des konkreten Projekts. Dies impliziert z.B. die Auswahl einer Alternative, falls mehrere Möglichkeiten im generischen Prozessmodell vorhanden sind (Bsp.: Soll die Aktivität des generischen Prozessmodells „Testen“ im konkreten Projekt durch „White-box-Test“ oder „Code Review“ realisiert werden). Außerdem werden die klassischen Aufgaben der Projektplanung unterstützt (Kostenschätzung, Zuweisung von Aufgaben, etc; eine genauere Übersicht über diesen Aspekt gibt [MDBG00]).

- Projektdaten:

Die Projektdaten enthalten alle Informationen, die während der Ausführung des Workflows erzeugt werden. Dazu gehören insbesondere die oben erwähnten Kontextinformationen.

Führt der Bearbeiter eine Aktivität aus und findet er relevantes Kontextwissen zu dieser Aktivität, das er auch für zukünftige Projekte als relevant erachtet, kann er die entsprechenden Ressourcen zu der Aktivität hinzufügen ([MH99] beschreibt ein Beispiel, in dem eine Softwareentwicklerin Ressourcen zum Thema Java zur Aktivität „Implementierung“ hinzufügt). Der Prozessverantwortliche entscheidet bei einer späteren Evaluation des Projekts, ob

die zusätzlichen Kontextinformationen in das generische Prozessmodell aufgenommen werden sollen und damit den Bearbeitern in zukünftigen Projekten zur Verfügung stehen. Durch diesen Zyklus wird der Aufbau und die Nutzung prozessbezogenen Wissens durch die Bearbeiter unterstützt. Allerdings können auch bei diesem System die Bearbeiter nicht direkt, d.h. ohne die nachträgliche Evaluierung des Projekts durch den Prozessverantwortlichen, kooperativ Wissen aufbauen und nutzen.

Wie Bramble, KnowMore und EULE2 zielt auch MILOS primär auf die Bereitstellung von Kontextwissen zu Aktivitäten. Im Gegensatz zu den genannten Systemen wird dieses Kontextwissen in MILOS aber nur teilweise bereits zum Modellierungszeitpunkt beschrieben (durch die genannten Abfragen, die zu einer Aktivität angegeben werden können). Darüber hinaus kann der Bearbeiter während der Durchführung von Aktivitäten relevantes Kontextwissen in Form von Informationen angeben, die durch die Phase der Evaluation ggf. in das generische Prozessmodell eingehen. Somit ist der Aspekt des kooperativen Wissensaufbaus durch die Bearbeiter bei MILOS im Vergleich zu anderen Ansätzen stärker ausgeprägt.

3.9 PROMOTE

Bei PROMOTE handelt es sich um ein EU-Projekt mit dem Ziel, ein integriertes Framework für prozessbezogenes Wissensmanagement zu erstellen und die entwickelte Methodologie anhand des Einsatzes eines zu entwickelnden Produkts (ebenfalls mit dem Namen PROMOTE) bei Finanzdienstleistern zu evaluieren [KT00]. Der PROMOTE-Ansatz kombiniert Wissensprozesse und Geschäftsprozesse sowohl durch Methoden zur Modellierung von beiden Prozessstypen und zur Spezifikation von Beziehungen zwischen Wissensprozessaktivitäten und Geschäftsprozessaktivitäten als auch durch Möglichkeiten einer entsprechend integrierten Durchführungsunterstützung der Prozesse. Abbildung 20 zeigt die PROMOTE-Methodologie. Wir wollen uns bei der folgenden Beschreibung der relevanten Aspekte auf die Phasen der Modellierung und Implementierung beschränken:

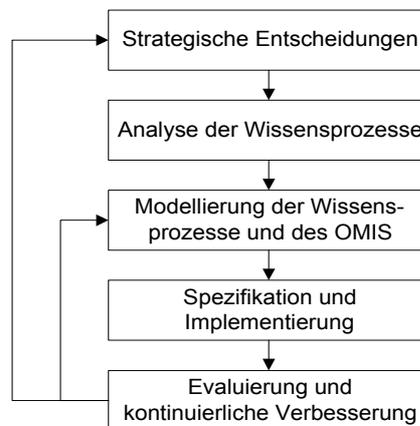


Abbildung 20: Die PROMOTE-Methodologie [KT00]

Eines der Hauptziele des Projekts besteht in der Entwicklung einer Beschreibungssprache für Wissensprozesse, wie z.B. Akquisition, Suche und Navigation oder Bewahrung. Dabei können Elemente klassischer Prozessmodellierungssprachen nur eingeschränkte Verwendung finden, da diese Sprachen im Allgemeinen eine zu starke Sequentialität der Prozessdurchführung voraussetzen.

Für die Spezifikation und Implementierung des PROMOTE-Produkts steht die Frage der geeigneten Verknüpfung von Wissensprozessen und Geschäftsprozessen im Vordergrund. Da die Relevanz z.B. eines Dokuments für bestimmte Bearbeitungssituationen nicht allein aus seinem Inhalt abgeleitet werden kann, wird insbesondere der Betrachtung des zugehörigen Prozesskontexts große Bedeutung eingeräumt, um die richtigen Informationen an die geeigneten Bearbeiter zu leiten. Dazu wurde eine Repräsentation von Wissensaktivitäten („knowledge activities“ [KT00]), d.h. Aktivitäten eines Geschäftsprozesses, die hohe Wissensbedarfe besitzen oder Wissen erzeugen, entwickelt. Diese Repräsentation enthält Kontextinformationen zur Aktivität, eine Abbildungsfunktion und eine Ausführungsbedingung. Ist die Wissensaktivität mit einem oder mehreren Wissensprozessen verknüpft (z.B. „Experten kontaktieren“), wird zur Laufzeit ein Wissensprozess entweder automatisch durch das System oder manuell durch den Bearbeiter ausgewählt. Die Kontextinformationen (Name des Bearbeiters, seine Rolle, Beschreibung der Aktivität, etc.) werden durch die Abbildungsfunktion auf entsprechende Attribute des ausgewählten Wissensprozesses abgebildet. Der Wissensprozess setzt dann in diesem Beispiel die gegebenen Informationen in eine E-Mail um und benachrichtigt den geeigneten Experten.

Bei den Arbeiten im Projekt PROMOTE handelt es sich um „work in progress“, die noch nicht abschließend beurteilt werden kann. Der Ansatz der integrierten Modellierung und Durchführung von sowohl Geschäftsprozessen als auch Wissensprozessen geht in seiner Komplexität über die oben dargestellten Arbeiten hinaus.

3.10 Zusammenfassung und Bewertung der Ansätze

Die Beschreibung der Systeme wurde insbesondere unter Berücksichtigung der schon in Abschnitt 3.1 erläuterten Fragestellungen vorgenommen:

- Welches Ziel wird mit einer wissensbezogenen Erweiterung des WFMS verfolgt?
- Welches Wissen wird verwaltet?
- Worauf bezieht sich das verwaltete Wissen?
- Welche Operationen auf dem verwalteten Wissen stehen dem Bearbeiter zur Verfügung?

Wir werden in diesem Abschnitt eine zusammenfassende Betrachtung der Systeme unter den genannten Aspekten geben.

In Bezug auf die Zielsetzung der wissensbezogenen Erweiterung eines WFMS lassen sich drei Hauptziele unterscheiden: Dabei handelt es sich um die kontinuierliche Prozessverbesserung (WorkBrain, Workware, PROMOTE), die Unterstützung der flexiblen Konfiguration und Ausführung von Geschäftsprozessen (WorkBrain, Workware, MILOS) und die Unterstützung wissensintensiver Aktivitäten (Bramble, KnowMore, OntoBroker, EULE2, MILOS, PROMOTE).

Bei der Beschreibung des verwalteten Wissens kann zwischen Prozesswissen und Fachwissen unterschieden werden. Während die Ansätze zur Unterstützung der kontinuierlichen Prozessverbesserung und flexiblen Konfiguration von Geschäftsfällen das Prozesswissen in den Vordergrund stellen, zielt die Unterstützung wissensintensiver Aktivitäten hauptsächlich auf die Bereitstellung relevanten Fachwissens (eine Ausnahme bildet Bramble; hier wird lediglich Prozesswissen zur Verfügung gestellt). Das Fachwissen wird in den verschiedenen Ansätzen allerdings ganz unterschiedlich repräsentiert: Während z.B. EULE2 Fachwissen in einer eigenen formalen Repräsentation speichert, um eine automatische Auswertung sicherstellen zu können, erlauben z.B. MILOS und WorkBrain die Verwaltung von unstrukturiertem Wissen durch die Verknüpfung von beliebigen Dokumenten mit Aktivitäten (MILOS) bzw. die Explizierung von Fachwissen durch ein Diskussionsforum (WorkBrain).

In Bezug auf den Kontext des verwalteten prozessbezogenen Fachwissens beschränken sich die Systeme zur Unterstützung wissensintensiver Aktivitäten gemäß ihrer Zielsetzung auf das Wissen, das direkt mit einzelnen Aktivitäten zusammenhängt. WorkBrain dagegen fokussiert stärker den Aspekt des geschäftsfallbezogenen Wissens. Keines der Systeme bietet aber einen konzeptionell überzeugenden Ansatz, um prozessbezogenes Wissen insgesamt in Beziehung zu den relevanten Objekten der Prozess- und Organisationsstruktur setzen zu können. Dies ist aber notwendig, um prozessbezogenes Fachwissen (in Form von Dokumenten, etc.) zu Prozessen, Geschäftsfällen, Aktivitäten, Rollen, Bearbeitern, etc. verwalten zu können und somit ein breites Spektrum an unterschiedlichen Kontexten und Nutzungssituationen abbilden zu können.

Auch die Operationen, die dem Bearbeiter auf dem verwalteten Wissen zur Verfügung stehen, und damit die Möglichkeiten der Wissensverteilung und Wissensbewahrung oder lediglich der Wissensnutzung sind in den einzelnen Systemen sehr unterschiedlich ausgeprägt. Bei den Systemen KnowMore, Ontobroker und EULE2 werden die Wissensbedarfe, die den Bearbeiter bei der Durchführung einer bestimmten Aktivität im Prozess betreffen, bereits zum Modellierungszeitpunkt quasi als parametrisierbare Abfrage festgelegt. Bei der Durchführung der Aktivität werden dann die Wissensbedarfe durch die Attributwerte des aktuellen Geschäftsfalls parametrisiert und die Ergebnisse dem Bearbeiter angeboten. Der Bearbeiter kann nun diese Hilfestellung des Systems in Anspruch nehmen und sich die entsprechenden Informationen vom System anzeigen lassen. Dagegen besteht für ihn nicht die Möglichkeit, Erfahrungen oder relevante Dokumente integriert in die Prozessbearbeitung mit der aktuellen Aktivität (oder weiteren Objekten der Prozess- bzw. Organisationssicht) zu verknüpfen, um dieses Wissen den Bearbeitern zukünftiger Geschäftsfälle zur Verfügung zu stellen und damit zu einem geschäftsfallübergreifenden Wissenstransfer beizutragen. Lediglich Systeme wie MILOS und WorkBrain bieten Ansätze sowohl zur Explizierung von Wissen direkt aus der Prozessbearbeitung heraus als auch zur Verknüpfung dieses Wissens mit Aktivitäten oder Geschäftsfällen. Wir werden später deutlich machen, dass diese Möglichkeiten allerdings nicht ausreichend sind, um wissensintensive Prozesse geeignet unterstützen zu können.

Die vorhandenen Systeme bilden also in Bezug auf ihre Zielsetzung und die damit verbundenen Funktionalitäten verschiedene Möglichkeiten wissensbezogener Erweiterungen von WFMS ab. Keines der Systeme ist aber nach unserer Meinung geeignet, die Klasse der wissensintensiven Prozesse, die wir im nächsten Kapitel vorstellen werden, geeignet zu unterstützen. Bei der konzeptionellen Ausgestaltung unseres Ansatzes werden wir daher die folgenden Schwerpunkte setzen, die unserer Meinung nach in den bestehenden Systemen nicht ausreichend betrachtet wurden:

- Verwaltung eines breiten Spektrums an prozessbezogenem Wissen:

Unter prozessbezogenem Wissen verstehen wir das Wissen, das für die Bearbeitung von (Geschäfts-)Prozessen relevant ist. Dabei unterscheiden wir primäres und sekundäres Wissen (exakte Definitionen gibt Abschnitt 5.2): Primäres Wissen ist für die Bearbeitung aller Geschäftsfälle relevant und wird vom Modellierer im Prozessmodell festgelegt. Zu primärem Wissen gehören somit u.a. Informationen über Aktivitäten und die Reihenfolge ihrer Bearbeitung, die zugehörigen Bearbeiter und Werkzeuge und die Dokumente, die jeweils in einer Aktivität benötigt oder bearbeitet werden. Darüber hinaus existiert mit sekundärem Wissen weiteres prozessbezogenes Wissen, das für die Bearbeitung von Aktivitäten in Prozessen wichtig ist, dessen Relevanz aber oftmals erst situativ in einer konkreten Bearbeitungssituation vom Bearbeiter festgestellt werden kann. Daher kann sekundäres Wissen, wie z.B. Gründe für Entscheidungen, Erfahrungen, Vorgaben bzgl. der Bearbeitung oder weitere Hintergrundinformationen, nicht sinnvollerweise im Prozessmodell definiert werden. Sowohl primäres als auch sekundäres Wissen muss von einem erweiterten WFMS unterstützt werden können.

- Unterstützung unterschiedlicher Funktionen und Charakteristik sekundären Wissens:

Wir werden die Funktion sekundären Wissens für die Prozessbearbeitung und die Charakteristik sekundären Wissens genauer untersuchen und konzeptionell unterstützen. Dabei sind u.a. Fragestellungen zu betrachten, unter welchen prozessbezogenen Bedingungen Wissen für den Bearbeiter relevant ist und für welche Ereignisse der Prozessdurchführung Informationen aktiv vom System bereitgestellt werden sollen.

- Strukturierung sekundären Wissens anhand des Bearbeitungskontexts:

Wir werden eine Strukturierung prozessbezogenen Wissens vorschlagen, die sich an den Objekten der Prozess- bzw. Organisationsstruktur orientiert und den Bearbeiter in die Lage versetzt, Wissen zu dokumentieren oder aufzufinden, das beispielsweise relevant für die Durchführung einer bestimmten Aktivität ist, sich auf den gesamten Prozess bezieht, lediglich für die Durchführung eines bestimmten Geschäftsfalls relevant ist oder in Beziehung mit einer bestimmten Rolle steht. Zu diesem Zweck werden wir das Konzept der Kontexte einführen.

- Kooperativer Aufbau und Bewahrung sekundären Wissens:

Bearbeiter sollten nicht nur in der Lage sein, zum Modellierungszeitpunkt definierte kontextbezogene Sichten auf den Wissensbestand abzufragen, sondern sie sollten auch zu einer Wissensbewahrung und zu einem Wissenstransfer beitragen können. Daher werden wir unseren Ansatz an einem kooperativen Aufbau und einer kooperativen Bewahrung von sekundären Wissen ausrichten.

Die hier skizzierten Anforderungen werden wir in Kapitel 5 konkretisieren. Als Grundlage für diese Konkretisierung erläutern wir im folgenden Kapitel zunächst den Begriff des „wissensintensiven Prozesses“.

4 Wissensintensive Prozesse

In Kapitel 3 wurden verschiedene Ansätze zur Integration von Workflow Management und Wissensmanagement mit unterschiedlichen Zielsetzungen und Konzepten vorgestellt. Um nun für diese Arbeit exakte Anforderungen an eine wissensorientierte Erweiterung von WFMS stellen zu können, muss zunächst geklärt werden, welche Art von Prozessen mit Hilfe erweiterter WFMS unterstützt werden soll. In diesem Abschnitt führen wir daher basierend auf Projekterfahrungen die Klasse der wissensintensiven Prozesse ein und diskutieren deren Eigenschaften (Abschnitt 4.1). Ein Beispiel aus einem Anwenderprojekt dient zur weiteren Illustration dieser Prozessklasse (Abschnitt 4.2).

4.1 Begriffsbildung

Wie in vorherigen Kapitel dargestellt wurde, werden die Begriffe „Wissensprozess“ und „wissensintensiver Prozess“ von den einzelnen Autoren bzw. in den dargestellten Konzepten unterschiedlich verwendet. Wir stellen in diesem Abschnitt unsere eigene Sichtweise und Charakterisierung von wissensintensiven Prozessen aus einer Anwendungssicht heraus vor. Dies erfolgt unter Berücksichtigung des augenblicklichen Einsatzspektrums von WFMS und des Begriffs der Wissensarbeit (Abschnitt 2.2.4). Nach einer ersten grundlegenden Darstellung der Eigenschaften wissensintensiver Prozesse (Abschnitt 4.1.1) werden wir zwei Fragestellungen erläutern, die einer genaueren Diskussion bedürfen und die die Klasse der wissensintensiven Prozesse weiter konkretisieren: Es sind erstens Probleme bei einem Transfer von Wissen zwischen unterschiedlichen Prozessen zu untersuchen (Abschnitt 4.1.2), und zweitens ist der Aspekt der Bewahrung prozessbezogenen Wissens zu betrachten (Abschnitt 4.1.3). Schließlich gibt Abschnitt 4.1.4 eine Zusammenfassung der Eigenschaften wissensintensiver Prozesse.

4.1.1 Einordnung und grundlegende Charakterisierung

Wie in Abschnitt 2.1.4 dargestellt, werden WFMS heutzutage hauptsächlich zur Unterstützung von Routineprozessen eingesetzt, also Prozessen, die einen hohen Anteil an Sachbearbeitertätigkeiten beinhalten. Die prozessbezogenen Informationen können bei Routineprozessen zum Modellierungszeitpunkt festgelegt werden, eine solide Unterstützung durch heutige Workflow-Management-Technologie ist damit gegeben. Andererseits rückt in der unternehmerischen Praxis der Begriff der Wissensarbeit immer stärker ins Zentrum der Betrachtung. Wie in Abschnitt 2.2.4 erörtert, ist aber Wissensarbeit für eine durchgängige Unterstützung durch Workflow Management nicht geeignet.

Die beiden Begriffe „Routineprozesse“ und „Wissensarbeit“ spannen also bzgl. der für diese Arbeit relevanten Fragestellungen ein Spektrum von Prozessen auf, das nur zu einem Teil workflow-geeignete Prozesse enthält. Aufbauend auf Projekterfahrungen werden im Weiteren Prozesse identifiziert, die prinzipiell workflow-geeignet sind, aber in Bezug auf die Anforderungen zur Verwaltung prozessbezogenen Wissens über die Möglichkeiten heutiger WFMS hinausgehen. Diese Prozesse bezeichnen wir als wissensintensive Prozesse. Eine Erweiterung von WFMS um die Möglichkeiten zur Unterstützung wissensintensiver Prozesse verspricht eine signifikante Ausweitung des Einsatzspektrums von WFMS (Abbildung 21).



Abbildung 21: Spektrum von Prozessen zwischen Routineprozessen und Wissensarbeit

Davenport beschreibt in [DJB96], dass Aktivitäten, die die Akquisition, Erstellung, Verbreitung und Anwendung von Wissen zum Inhalt haben, mehr und mehr in den Kernkompetenzen und in den Kerngeschäftsprozessen von Unternehmen sichtbar werden. Wir gehen bei der Definition wissensintensiver Prozesse schrittweise vor und führen daher zunächst in Anlehnung an die Beschreibung von Davenport den Begriff der wissensintensiven Aktivität ein. Der Begriff der wissensintensiven Aktivität dient später (Definition 11) zur Charakterisierung wissensintensiver Prozesse.

Definition 10: Wissensintensive Aktivität

„Eine wissensintensive Aktivität ist gekennzeichnet durch flexible, nicht planbare Wissensbedarfe, unterschiedliche Ergebnisse und einen überdurchschnittlich hohen Kommunikations- und Kooperationsbedarf bei der Durchführung der Aktivität.“

Im Folgenden werden diese Eigenschaften wissensintensiver Aktivitäten erläutert:

- Flexible, nicht planbare Wissensbedarfe:

In vielen Geschäftsfällen sind nicht nur die im Prozessmodell festgelegten Prozessmodell Dokumente zur Bearbeitung von Aktivitäten notwendig. Weitere Wissensbedarfe existieren, sie werden aber von den Bearbeitern größtenteils erst bei der Geschäftsfallbearbeitung identifiziert und sind abhängig von den Gegebenheiten des konkreten Geschäftsfalls. Die Relevanz von derartigen Wissensbeständen kann also nicht zum Modellierungszeitpunkt festgelegt werden. Zusätzlich fragen Bearbeiter oftmals Wissen nach, das zum Zeitpunkt der Modellierung noch gar nicht existiert, sondern erst während der Durchführung ggf. über eine Vielzahl von Geschäftsfällen hinweg aufgebaut wird.

Flexible, nicht planbare Wissensbedarfe treten z.B. oftmals bei Entscheidungen auf. Dabei ist einerseits ein Teil des benötigten Wissen stets gleich (wie z.B. Antragsformulare, etc.), andererseits wird in Abhängigkeit von der Komplexität der im aktuellen Geschäftsfall zu treffenden Entscheidung weiteres Wissen benötigt, wie z.B. Erfahrungen, Gründe für in ähnlichen Geschäftsfällen getroffene Entscheidungen oder Zusatzinformationen in Form von Checklisten, etc. Handelt es sich dagegen um eine Entscheidung von geringer Komplexität, ist dieses zusätzliche Wissen nicht von Bedeutung. Die Entscheidung über die Relevanz bestimmter Wissensbestände kann also in den meisten Fällen allein der Bearbeiter unter Berücksichtigung seiner konkreten Bearbeitungssituation treffen.

- Unterschiedliche Ergebnisse:

Die Ergebnisse von wissensintensiven Aktivitäten können ebenfalls nicht vollständig im Voraus festgelegt werden. Ergebnisse werden teilweise auf verschiedene Arten dokumentiert. Insbesondere entstehen neue Ergebnisse oftmals in Abhängigkeit der Eigenschaften des

konkreten Geschäftsfalls. Diese zusätzlichen Ergebnisse einer wissensintensiven Aktivität können von ganz unterschiedlichem Typ sein: Sonderfälle und Ausnahmen bedingen beispielsweise oftmals die Erstellung zusätzlicher Dokumente. Außerdem kann es gerade bei komplexen Aktivitäten erforderlich sein, Erfahrungen und bei der Bearbeitung gewonnenes Wissen zu dokumentieren oder Gründe für Entscheidungen festzuhalten, oder es können in Einzelfällen Feedback oder Verbesserungsvorschläge als Ergebnis einer Aktivität auftreten.

- Überdurchschnittlich hoher Kommunikations- und Kooperationsbedarf:

Wissensintensive Aktivitäten bedingen oftmals auf Grund ihrer Komplexität und der damit zusammenhängenden hohen Spielräume der Bearbeiter die Kooperation und den Wissensaustausch mit weiteren Personen. Das Einschalten von Experten oder der Rückgriff auf Erfahrungen von Mitarbeitern, die ähnliche Situationen bereits bearbeitet haben, sind hier typische Beispiele. Im Unterschied dazu werden die klassischen Sachbearbeiteraktivitäten zumeist von einer einzigen Person in relativ kurzer Zeit ohne weiteren Kommunikations- und Kooperationsbedarf ausgeführt.

Mit Hilfe des Begriffs der wissensintensiven Aktivität kann nun eine Klasse von Prozessen definiert werden, die einerseits prinzipiell workflow-geeignet ist, andererseits aber in Bezug auf die Anforderungen zur Verwaltung prozessbezogenen Wissens, wie oben dargestellt, über die Eigenschaften von heutigen workflow-geeigneten Prozessen hinaus geht. Dabei sind wissensintensive Aktivitäten in der Praxis oftmals in Prozesse eingebettet, die einen erheblichen Anteil an Sachbearbeitertätigkeiten besitzen.

Definition 11: Wissensintensiver Prozess

„Ein wissensintensiver Prozess ist ein Prozess, der sowohl eine oder mehrere wissensintensive Aktivität(en) enthält als auch - abgesehen von dem Aspekt der Verwaltung prozessbezogenen Wissens - zur Unterstützung durch Workflow Management geeignet ist.“

Wissensintensive Prozesse können in Bezug auf die in Abschnitt 2.1.4 skizzierten Kriterien als Routineprozesse angesehen werden. Lediglich das Kriterium der „ähnlichen Ausführung“ ist in Bezug auf den Aspekt prozessbezogenen Wissens nicht erfüllt, da dieses Kriterium im Widerspruch zur obigen Definition der wissensintensiven Aktivität steht.

Wissensintensive Prozesse sind darüber hinaus durch einen hohen Wissenstransfer sowohl innerhalb von Geschäftsfällen als auch zwischen Geschäftsfällen gekennzeichnet. Der Austausch von prozessbezogenem Wissen wird in heutigen WFMS im Allgemeinen nur innerhalb eines Geschäftsfalls unterstützt und z.B. durch eine elektronische Umlaufmappe abgewickelt, die beim Fortschreiten des Geschäftsfalls von einem Bearbeiter zum nächsten gesandt wird (Abschnitt 2.1.3). In wissensintensiven Prozessen treten allerdings weitere Wissenstransfers auf: So kann es z.B. innerhalb eines Geschäftsfalls notwendig sein, auf Erfahrungen oder Dokumente zurückzugreifen, die in vorherigen Aktivitäten gewonnen wurden, ohne dass dies im Prozessmodell festgeschrieben ist. Darüber hinaus finden Wissenstransfers auch zwischen Geschäftsfällen statt, indem auf Wissen, das in einem Geschäftsfall erzeugt wurde, in späteren Geschäftsfällen zurückgegriffen wird. Dieser Transfer prozessbezogenen Wissens ist zu einem erheblichen Anteil in der Struktur des Prozesses begründet: Insbesondere wissensintensive Aktivitäten erfordern oftmals einen Austausch von Wissen orthogonal zu den Geschäftsfällen.

Abbildung 22 zeigt dazu beispielhaft die Bearbeitung von Geschäftsfällen und den zugehörigen Transfer von Wissen. Neben der Wissensverteilung innerhalb des Geschäftsfalls gemäß dem Prozessmodell (durch Pfeile zwischen den Aktivitäten innerhalb der Geschäftsfälle dargestellt), die durch z.B. eine elektronische Umlaufmappe eines WFMS unterstützt werden kann, sind weitere Wissenstransfers, wie oben beschrieben, ersichtlich.

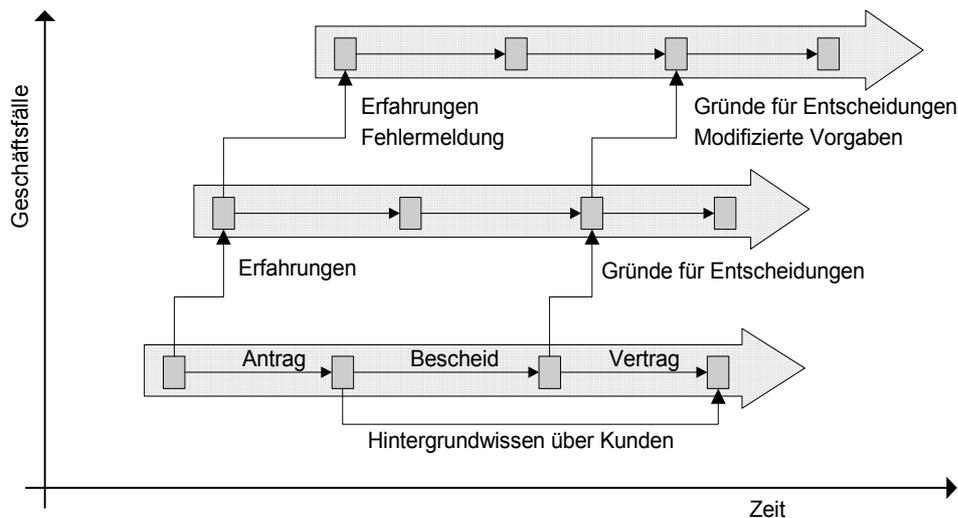


Abbildung 22: Wissenstransfer innerhalb und zwischen Geschäftsfällen

Wir haben in diesem Abschnitt eine erste grundlegende Definition der Klasse der wissensintensiven Prozesse gegeben, die als Basis für eine praxisrelevante Erweiterung von WFMS dienen soll. Dabei sind allerdings noch zwei wichtige Aspekte nicht ausreichend erörtert worden, die in den folgenden beiden Abschnitten eine stärkere Fokussierung des Begriffs des wissensintensiven Prozesses bedingen:

Wissenstransfers zwischen Geschäftsfällen sind nach der bisherigen Beschreibung nicht notwendigerweise auf Geschäftsfälle eines Prozesses beschränkt, sondern können sich auch auf Geschäftsfälle unterschiedlicher Prozesse erstrecken. Die mit diesem Aspekt zusammenhängenden Fragestellungen beleuchtet Abschnitt 4.1.2. Bei der Diskussion der Anforderungen an eine Erweiterung von WFMS zur Unterstützung wissensintensiver Prozesse ist die Bewahrung prozessbezogenen Wissens zu berücksichtigen. Eine geeignete Einschränkung der Klasse wissensintensiver Prozesse liefert hierzu einen Lösungsansatz (Abschnitt 4.1.3).

4.1.2 Wissenstransfer zwischen Geschäftsfällen unterschiedlicher Prozesse

Ein Transfer von Wissen zwischen Geschäftsfällen unterschiedlicher Prozesse macht in der unternehmerischen Praxis in verschiedenen Anwendungsfällen Sinn. Beispielsweise werden oftmals bestimmte Kundendaten in einem Prozess erfasst, die zumindest teilweise für die Bearbeitung weiterer Prozesse relevant sind. Denkbar ist auch, dass Aktivitäten, die in unterschiedlichen Prozessen modelliert sind, in Bezug auf die ihnen jeweils zugehörige Aufgabe inhaltliche

Überschneidungen besitzen und damit auch Wissen, das z.B. bei der Bearbeitung einer Aktivität entsteht, zur Durchführung einer weiteren Aktivität relevant sein kann. Dieser inhaltliche Bezug zwischen verschiedenen Prozessen kann in einigen Fällen sehr eng und damit offensichtlich sein. So ist es unmittelbar einleuchtend, dass bestimmte Wissensbestände, die im Rahmen einer Vertragserstellung erzeugt werden, für eine spätere Vertragsverlängerung mit diesem Kunden relevant sein können. Allerdings existieren darüber hinaus oftmals inhaltliche Zusammenhänge zwischen Prozessen, die weitaus schwieriger zu erkennen sind und deren Identifikation detaillierte Kenntnisse der beteiligten Prozesse erfordert. Diese Problematik wirft aber unmittelbar die Frage auf, wie inhaltliche Zusammenhänge zwischen Prozessen identifiziert werden können und wie der prozessübergreifende Transfer von Wissen somit systematisch unterstützt werden kann. Im einzelnen sind insbesondere die folgenden Aspekte zu berücksichtigen:

- Rekonstruktion der Fachbegriffe:

Unterschiedliche Prozesse aus ggf. sogar unterschiedlichen Abteilungen oder Geschäftsbereichen werden im Allgemeinen durch verschiedene Modellierer abgebildet. Dadurch ergeben sich Probleme, bei der Abbildung des Anwendungsbereichs eine einheitliche konsistente Begriffswelt aufzubauen [JBS97]. Insbesondere die Verwendung von Synonymen (umfangs- und inhaltsgleiche Begriffe, deren Bezeichner gegeneinander ausgetauscht werden können) und Homonymen (umfangs- und inhaltsverschiedene Begriffe mit einem gemeinsamen Bezeichner) führt zu unscharfen, unkorrekten oder widersprüchlichen Aussagen. Darüber hinaus existiert das Problem der Vagheit, wenn Begriffe nicht präzise genug benannt werden: Beispielsweise finden sich in Prozessmodellen oftmals Rollenbezeichner, wie „Abteilungsleiter“ oder „Sekretärin“, oder Aktivitäten, wie „Antrag prüfen“ oder „Eskalation“. Die Bedeutung dieser Bezeichner ist aber zumeist auf den spezifischen Prozess beschränkt. Eine Eskalation einer Reisekostenabrechnung ist im Hinblick auf das benötigte Wissen grundverschieden von einer Eskalation einer Investitionsentscheidung. Die Notwendigkeit für einen prozessübergreifenden Wissenstransfer kann also nicht aus einer gleichen Bezeichnung von z.B. Aktivitäten abgeleitet werden. Ein Ansatz zur Lösung kann im Aufbau von Ontologien bestehen (Abschnitt 2.3), die eine Basis für ein unternehmensweites gemeinsames Vokabular darstellen, das von allen Modellierern in konsistenter Weise genutzt wird. Der Aufbau und die Wartung von Ontologien stellt allerdings einen schwierigen und zeitaufwändigen Prozess dar.

- Wartung der prozessübergreifenden inhaltlichen Beziehungen:

Die Wartung der prozessübergreifenden inhaltlichen Beziehungen, die einen Wissenstransfer zwischen Prozessen begründen, stellt insbesondere deshalb ein Problem dar, da in einer dynamischen Organisation Prozesse stets Veränderungen unterworfen sind. Existierende Prozesse werden im Rahmen eines kontinuierlichen Verbesserungsprozesses (Abschnitt 2.1.2) modifiziert, und durch strategische Neuausrichtungen oder Optimierungen des Geschäfts werden neue Prozesse erstellt bzw. vorhandene Prozesse eingestellt. In jedem dieser Fälle ist nun der entsprechende Modellierer verpflichtet, inhaltliche Beziehungen zu anderen Prozessen aktuell zu halten. Damit sind zeitaufwändige Abstimmungsprozesse unvermeidbar.

- Problem des Kontexts bei einem Transfer von Wissen:

Ein weiterer Aspekt betrifft das Problem des Kontexts beim Transfer von Wissen zwischen Prozessen und damit zwischen Bearbeitern aus ggf. unterschiedlichen Organisationseinheiten mit unterschiedlichen Aufgaben. Erfolgreicher Wissenstransfer bedingt stets das Verstehen auf Seiten des Empfängers. Dies bedeutet aber, dass der angemessene Grad von Kontextinformation ermittelt werden muss, damit Sender und Empfänger „die gleiche Sprache sprechen“, also ausgetauschtes Wissen für den Empfänger weder verwirrende Details ohne erkennbaren Zweck beinhaltet noch zu allgemein und damit ebenfalls unbrauchbar ist (Abschnitt 2.2.3). Wenn Wissen also zwischen Prozessen, die thematisch, also in Bezug auf die jeweils durchzuführende Aufgabe, kaum verwandt sind, ausgetauscht werden soll, muss das Wissen für den jeweiligen Empfänger aufbereitet werden. Dabei handelt es sich im Allgemeinen um eine komplexe Aufgabe, die zumindest teilweise manuell durchgeführt werden muss und die daher mit hohen Aufwänden verbunden ist.

Die bisherigen Betrachtungen legen den Schluss nahe, dass eine Einschränkung der Klasse der wissensintensiven Prozesse anzustreben ist, um die Komplexität der geschilderten Probleme in Grenzen zu halten, wobei allerdings die Praxistauglichkeit des Ansatzes gewährleistet bleiben muss. Ein Wissenstransfer zwischen unterschiedlichen Prozessen soll daher nicht vollkommen ausgeschlossen werden. In diesem Sinne beschränken wir uns im Folgenden darauf, den Austausch von Wissen zwischen wissensintensiven Prozessen nur dann in die weiteren Betrachtungen einzubeziehen, wenn diese Prozesse sehr starke Abhängigkeiten in Bezug auf sowohl die jeweils durchzuführende Aufgabe als auch die beteiligten Rollen und Organisationseinheiten besitzen. Dazu führen wir den Begriff der Prozessgruppe ein.

Definition 12: Prozessgruppe

„Eine Prozessgruppe ist eine Menge von Prozessen, die untereinander einen starken inhaltlichen Bezug aufweisen und durch ähnliche Gruppen von Bearbeitern durchgeführt werden.“

Die beiden Eigenschaften einer Prozessgruppe werden im Folgenden erläutert:

- Starker inhaltlicher Bezug:

Die Prozesse einer Prozessgruppe weisen einen starken inhaltlichen Bezug auf. Dies kann z.B. daran ersichtlich sein, dass die Prozesse einer Prozessgruppe jeweils für verschiedene Phasen im Lebenszyklus von gemeinsamen Objekten zuständig sind (z.B. Vertrag erstellen, Vertrag ändern und Vertrag auflösen) oder zusammen ein homogenes Spektrum von Dienstleistungen in der Form aufbauen, dass die zentralen Objekte, die die einzelnen Prozesse jeweils behandeln, inhaltlich stark verwandt sind. Ein diesbezügliches Beispiel stellen die Kerngeschäftsprozesse in einem Bauordnungsamt, wie Bauvoranfrage, Bauantrag, Bauantrag für Werbeanlagen, Abbruchantrag, etc. dar. Auf Grund der starken inhaltlichen Abhängigkeiten werden alle Prozesse einer Prozessgruppe idealerweise von einem Modellierer modelliert. Bei einer Einschränkung von Wissenstransfers auf Prozesse einer gemeinsamen Prozessgruppe werden die ersten beiden aufgeführten Probleme signifikant reduziert, da erstens eine eindeutige Namensgebung für die Entitäten der betroffenen Prozessmodelle auf wesentlich einfachere Weise durchgeführt werden kann. Zweitens müssen bei einer Modifikation von Prozessen neue inhaltliche Bezüge, die Wissenstransfers zwischen Prozes-

sen notwendig machen, lediglich innerhalb der Prozessgruppe identifiziert werden. Da diese Aufgaben im Allgemeinen in der Verantwortung eines einzigen Modellierers liegen, entfallen zeitraubende Abstimmungsprozesse innerhalb der gesamten Organisation.

- Ähnliche Gruppen von Bearbeitern:

Ein weiteres Kriterium für die Identifikation von Prozessgruppen besteht darin, dass die einzelnen Prozesse von ähnlichen Bearbeitergruppen durchgeführt werden. Unter ähnlichen Gruppen verstehen wir, dass entweder zum großen Teil dieselben Personen an mehreren Prozessen beteiligt sind (Bsp. Rundumsachbearbeitung in einer Versicherung) oder wenn dies nicht der Fall ist, zumindest starke Überschneidungen bzgl. der Qualifikation und des Know-how bei den Bearbeitern verschiedener Prozesse existieren. Durch diese Eigenschaft wird die Problematik entschärft, dass für einen Wissenstransfer zwischen unterschiedlichen Prozessen eine aufwändige Aufbereitung des Wissens notwendig ist, die die unterschiedlichen Aufgaben, Kenntnisse und Qualifikationen der einzelnen Bearbeiter berücksichtigen muss (Problem des Kontexts beim Transfer von Wissen). Außerdem trägt dieses Kriterium zu einer vereinfachten Wissensbewahrung durch die Bearbeiter bei (siehe nächster Abschnitt).

Wie oben dargestellt wurde, führt die Unterstützung von Wissenstransfers zwischen unterschiedlichen Prozessen zu einer Menge von Problemen, die folgende Einschränkung impliziert: Wissenstransfers zwischen Prozessen werden lediglich innerhalb von Prozessgruppen betrachtet. Durch diese Restriktion werden die geschilderten Probleme signifikant vereinfacht bzw. vermieden, ohne dass die Praxistauglichkeit des Ansatzes in Frage gestellt werden muss.

4.1.3 Bewahrung prozessbezogenen Wissens

Die systematische Wissensbewahrung (Abschnitt 2.2.2), die durch eine Selektion des Bewahrungswürdigen, angemessene Speicherung und regelmäßige Aktualisierung [PRR98] das relevante organisationale Wissen verwaltet, stellt ein erhebliches Problem für das Wissensmanagement dar. Nach unserem Verständnis muss eine Bewahrung von Wissen durch eine inhaltliche Bewertung und Aktualisierung letztendlich in der Verantwortung von Menschen liegen, informationstechnische Systeme können hier lediglich unterstützende Funktionalitäten bieten. Daher müssen bei der Diskussion dieser Fragestellung neben technischen Aspekten auch organisatorische und motivationale Hindernisse berücksichtigt werden. Im Folgenden soll kurz dargestellt werden, welche Aspekte allgemein einer erfolgreiche Bewahrung von Wissen entgegen stehen:

- Wissensüberflutung der Bearbeiter:

In vielen Fällen ist die Menge der rechnergestützt angebotenen Informationen und des darüber hinaus als relevant erscheinenden Wissens für eine gegebene Fragestellung zu groß, um für den Bearbeiter eine sinnvolle Unterstützung darzustellen. Klassische Beispiele sind Anfragen bei Internet-Suchmaschinen, aber auch die Suche in OMIS oder Dokumenten-Management-Systemen liefert oftmals eine für den Bearbeiter unüberschaubare Menge von Informationen zurück. Die Relevanz und Aktualität dieser Informationen kann von einem Bearbeiter nicht in einem akzeptablen Zeitraum geprüft werden. Darüber hinaus stellt sich ein Gefühl der Wissensüberflutung ein, das für den einzelnen demotivierend wirkt.

- Fehlende Verantwortlichkeiten und Anreize für die Wissensbewahrung:

Die Anreiz- und Erwartungssituation bei der Wissensbewahrung unterscheidet sich fundamental von der bei der Bearbeitung klassischer Aufgaben in Prozessen. Der Nutzen z.B. der Aktualisierung eines Dokuments ist weniger durchschaubar, es fehlt ein interner oder externer Kunde, der auf ein Ergebnis wartet, und damit auch oftmals ein unmittelbares Feedback. Oftmals fehlen zusätzlich klare Verantwortlichkeiten, welche Mitarbeiter jeweils für die Bewahrung eines bestimmten Anteils des organisationalen Wissens verantwortlich sind. Kann dann nicht der Nutzen der Wissensbewahrung für konkrete Personen herausgestellt werden, sind Wissensmanagementprojekte zum Scheitern verurteilt.

- Seltener Zugriff auf Wissensbestände:

Ein weitere Problematik ergibt sich, wenn Wissen nur in seltenen Fällen benötigt und damit zugegriffen wird. Wird ein Wissensbestand nur sehr selten benötigt, ist die Wahrscheinlichkeit hoch, dass beim Zugriff ein großer Anteil des Wissens bereits veraltet und damit nicht mehr gültig ist. Selbst wenn also die oben geschilderten Probleme in einer Organisation bewältigt wurden, also jeweils eine überschaubare Menge von Wissen angeboten wird und sowohl Verantwortlichkeiten als auch Nutzen der Wissensbewahrung ersichtlich sind, treten in diesem Fall motivationale Probleme auf, da für eine konsequente Wissensbewahrung bei jedem Zugriff eine zeitaufwändige Aktualisierung eines großen Anteils des Wissensbestandes vorgenommen werden muss.

Das Problem der Wissensbewahrung kann teilweise durch eine geeignete Einschränkung der betrachteten Prozessklasse vereinfacht werden. Die beiden zuerst dargestellten Aspekte der Wissensüberflutung und der fehlenden Verantwortlichkeiten und Anreize für die Wissensbewahrung sollen durch geeignete Konzepte zur Strukturierung und Präsentation prozessbezogenen Wissens angegangen werden (eine weitere Erläuterung dieser Anforderung erfolgt in Abschnitt 5.3). Der zuletzt dargestellte Aspekt des seltenen Zugriffs auf Wissensbestände hängt dagegen direkt mit der Charakteristik der betrachteten Klasse von Prozessen zusammen und soll in diesem Abschnitt diskutiert werden.

Ein hoher Zeitaufwand des einzelnen Bearbeiters für die Wissensbewahrung aufgrund des insgesamt seltenen Zugriffs auf prozessbezogenes Wissen kann vermieden werden, indem lediglich Prozesse betrachtet werden, bei denen auf Grund ihrer Charakteristik ein häufiger Wissensaustausch innerhalb und zwischen Geschäftsfällen angenommen werden kann. Daher charakterisieren wir wissensintensive Prozesse zusätzlich durch folgende Merkmale:

- Häufige Ausführung:

Wissensintensive Prozesse werden häufig instanziiert. Dabei sind aber nicht ausschließlich Prozesse gemeint, die im klassischen Massengeschäft ablaufen. Eine Instanzierungshäufigkeit von z.B. mehreren Geschäftsfällen pro Woche erfüllt dieses Kriterium gleichermaßen. Durch diese Eigenschaft soll sichergestellt werden, dass bei Wissenstransfers zwischen Geschäftsfällen die Zeitdauer zwischen Erstellung bzw. Aktualisierung eines Wissensbestands und der nächsten Nutzung nicht dazu führt, dass ein Großteil des Wissens bereits nicht mehr gültig ist.

- Geringe bis mittlere Durchlaufzeit:

Für einen erfolgreichen Wissenstransfer innerhalb eines Geschäftsfalls ist die Durchlaufzeit ein wichtiger Parameter. Wird z.B. bei langlaufenden Prozessen, wie z.B. der Durchführung von Projekten, gegen Ende des Geschäftsfalls auf zu Beginn gewonnenes Wissen zurückgegriffen, entstehen auch hier mit einer hohen Wahrscheinlichkeit die geschilderten Probleme mangelnder Wissensaktualität. Unter Prozessen geringer bis mittlerer Durchlaufzeit verstehen wir Prozesse, bei denen die mittlere Durchlaufzeit höchstens im Bereich von Tagen bis wenigen Wochen liegt.

In Abbildung 23 wird schematisch die Ausführung eines häufig instanziierten Prozesses mit mittlerer Durchlaufzeit und eines selten instanziierten Prozesses mit hoher Durchlaufzeit gegenübergestellt. Es ist deutlich erkennbar, dass bei wissensintensiven Prozessen ein wesentlich häufigerer Zugriff auf die beteiligten Wissensbestände erfolgt.

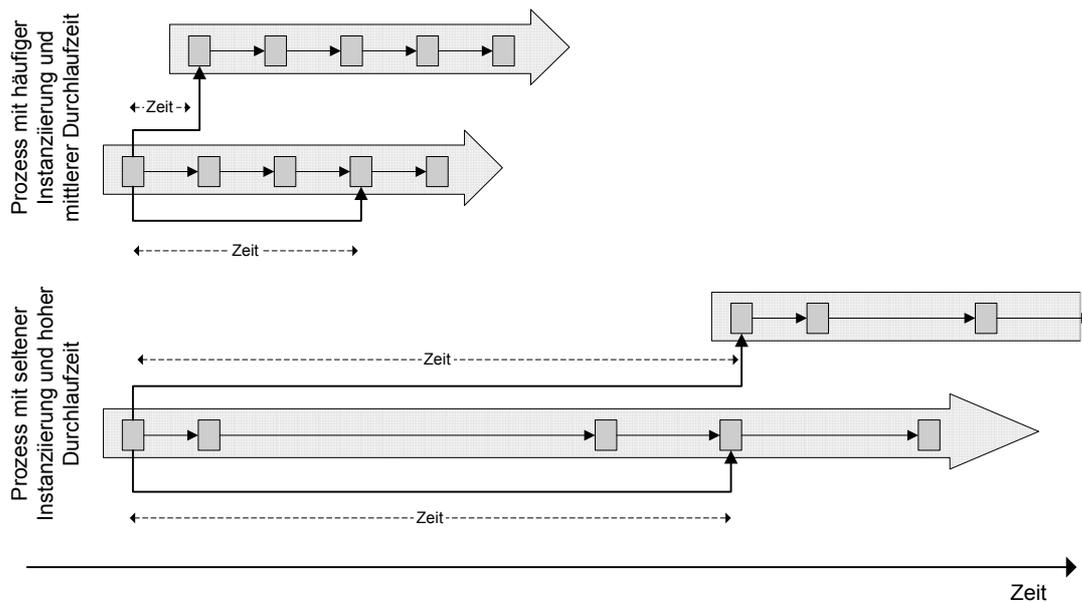


Abbildung 23: Zugriff auf Wissen bei Prozessen unterschiedlicher Ausführungshäufigkeit und Durchlaufzeit

Die in diesem Abschnitt vorgenommene Einschränkung der Klasse der wissensintensiven Prozesse trägt zu einer wesentlich einfacheren Bewahrung prozessbezogenen Wissens bei. Die Praxisrelevanz des Ansatzes wird durch diese Einschränkung allerdings kaum gemindert, da die überwiegende Zahl der von Unternehmen als workflow-relevant angesehenen Prozesse die obigen Kriterien ohnehin erfüllen (Abschnitt 2.1.4).

4.1.4 Zusammenfassung

Ausgehend von einer grundlegenden Charakterisierung wissensintensiver Prozesse erfolgte in den vorherigen beiden Abschnitten eine weitere Konkretisierung dieser Prozessklasse durch zusätzliche Eigenschaften. Im Folgenden sollen nun die Ergebnisse noch einmal kurz zusammengefasst werden.

Zunächst wurde der Begriff der wissensintensiven Aktivität eingeführt. Wissensintensive Aktivitäten sind charakterisiert durch:

- Flexible, nicht planbare Wissensbedarfe
- Unterschiedliche Ergebnisse
- Überdurchschnittlich hohen Kommunikations- und Kooperationsbedarf

Darauf aufbauend erfolgte die Einführung des Begriffs des wissensintensiven Prozesses. Wissensintensive Prozesse sind prinzipiell workflow-geeignet, sie enthalten eine oder mehrere wissensintensive Aktivitäten und sind darüber hinaus durch folgende Eigenschaften charakterisiert:

- Hoher Wissenstransfer sowohl innerhalb von Geschäftsfällen als auch zwischen unterschiedlichen Geschäftsfällen
- Häufige Ausführung
- Geringe bis mittlere Durchlaufzeit

Ein Wissenstransfer zwischen unterschiedlichen Prozessen soll nur unterstützt werden, falls die Prozesse zu einer gemeinsamen Prozessgruppe gehören. Unter einer Prozessgruppe verstehen wir eine Menge von Prozessen mit folgenden Eigenschaften:

- Starker inhaltlicher Bezug
- Ähnliche Gruppen von Bearbeitern

4.2 Beispiel für einen wissensintensiven Prozess: „Contract Management“

In diesem Abschnitt sollen die Ausführungen bzgl. der Beschreibung wissensintensiver Prozesse um ein Beispiel ergänzt werden. Damit soll einerseits eine anschauliche Konkretisierung der im vorherigen Abschnitt aufgeführten Eigenschaften gegeben werden, andererseits bietet das Beispiel einen Anknüpfungspunkt für das folgende Kapitel, da anhand der Darstellung Anforderungen an WFMS zur Unterstützung wissensintensiver Prozesse deutlich werden.

Das folgende Beispiel für einen wissensintensiven Prozess basiert auf dem Prozess der Anbahnung und Verwaltung von Verträgen mit neuen und bestehenden Kunden bei einem Kurier- und Expressdienstleister, im Folgenden „Contract Management“ genannt. Im Rahmen des Verbundprojekts MOVE¹ begleitete das Fraunhofer ISST das Unternehmen über einen Zeit-

raum von drei Jahren bei der Planung, Erstellung und Einführung einer Workflow-Management-Anwendung. Dabei wurde insbesondere eine detaillierte Erhebung und Modellierung des Prozesses durchgeführt.

Abbildung 24 gibt einen Überblick über den Prozess "Contract Management" anhand des zugehörigen Prozessmodells. Der Prozess, der hier in Bezug auf die durchzuführenden Aktivitäten und möglichen Alternativen der Bearbeitung vereinfacht dargestellt ist, dient zur Erstellung von Verträgen mit Kunden, die aufgrund ihres erhöhten Sendungsaufkommens bestimmte Sonderkonditionen erhalten. Grundsätzlich läuft der Prozess nach folgendem Muster ab: Der Kontakt mit dem Kunden wird durch einen der dezentral organisierten Verkäufer initiiert. Verkäufer und Kunde vereinbaren einen Termin, bei dem sie im Idealfall direkt eine Einigung über die dem Kunden zu gewährenden Rabatte erzielen. Der Verkäufer erstellt daraufhin eine Rabattanfrage, die die mit dem Kunden vereinbarte Übereinkunft in ein schematisiertes Ratenprofil überträgt. Dabei handelt es sich um eine komplexe Aufgabe, die dem Verkäufer bestimmte Spielräume lässt. Die Rabattanfrage wird an die Abteilung Sales, die zentral in der Hauptverwaltung des Unternehmens angesiedelt ist, geleitet. Dort entscheidet ein Mitglied des Preiskomitees über die Genehmigung oder Ablehnung der Raten, indem er eine Simulation des Ratenprofils durchführt. Unter besonderen Bedingungen erstellt der Mitarbeiter eine Entscheidungsvorlage, die von einer vorgesetzten Stelle genehmigt oder abgelehnt wird (Approval). In jedem Fall wird der zugehörige Rabattbescheid dem Verkäufer zugestellt, der nun bei Ablehnung seiner Rabattanfrage den Vorgang beendet. Wurde die Rabattanfrage genehmigt, erstellt die Abteilung Sales einen entsprechenden Vertrag. Dieser wird dann dem Kunden zugeleitet, und die dem Kunden gewährten Raten werden in ein zentrales Informationssystem eingetragen.

Das in Abbildung 24 dargestellte Prozessmodell enthält die Menge der Dokumente, die für eine Umsetzung des Prozesses mittels heute verfügbarer Workflow-Management-Technologie als geeignet und relevant angesehen wurden. Bei der detaillierten Betrachtung von konkreten Geschäftsfällen im Projekt fiel allerdings auf, dass situationsbezogen eine weitaus größere Menge von Dokumenten und Wissen für die Bearbeitung des Prozesses relevant ist bzw. in einer engen Beziehung zu dem Prozess steht. Dies hängt damit zusammen, dass es sich bei den drei Aktivitäten „Rabattierung mit Kunden verhandeln“, „Rabattanfrage erstellen“ und „Rabattanfrage entscheiden“ (in der Abbildung schattiert dargestellt) um wissensintensive Aktivitäten handelt, die damit zusätzliche Anforderungen an eine informationstechnische Unterstützung stellen, die von heutigen WFMS nur unzureichend erfüllt werden können. Die wissensintensiven Aspekte werden im Folgenden an einer detaillierteren Beschreibung eines einzelnen Geschäftsfalls illustriert:

-
1. Die Ergebnisse des Projekts wurden in [HSW98a, HSW98b, HSW99 und HSW01] veröffentlicht. Erfahrungsberichte aus dem Teilprojekt des hier beschriebenen Unternehmens finden sich in [GKSW98, GKLS98 und SGL98].

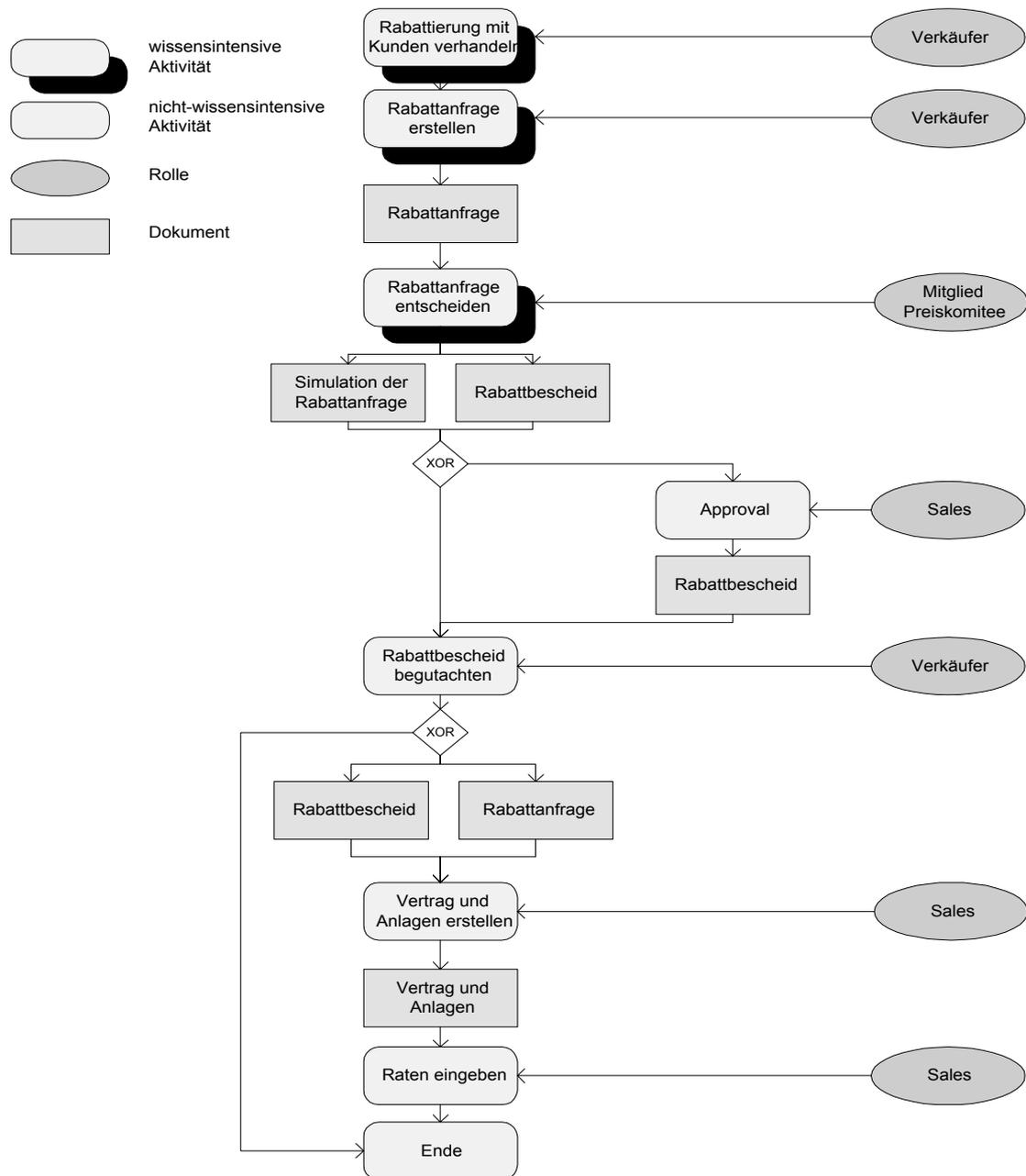


Abbildung 24: Der Prozess "Contract Management"

Der Verkäufer nimmt den Kontakt zum Kunden auf und vereinbart einen Termin, um über mögliche Rabattierungsalternativen zu verhandeln. Inzwischen hat der Marketing Manager eine neue Version des Dokuments zu möglichen Rabattierungen bei Großkunden erarbeitet. Da die Änderung dieser Richtlinie dem Verkäufer ggf. neue Spielräume bei der Verhandlung

mit dem Kunden eröffnet (oder diese auch beschränkt), soll dem Verkäufer das Dokument zum richtigen Zeitpunkt, also bei der Vorbereitung auf den Termin zugeleitet werden. Es handelt sich also um einen nicht planbaren Wissensbedarf für diese Aktivität.

Da der Kunde im Lauf des Gesprächs geäußert hatte, auch mit weiteren Wettbewerbern zu verhandeln, ist es wichtig, das im Kundengespräch gewonnene Wissen über Angebote der Wettbewerber auch weiteren Personen im Unternehmen verfügbar zu machen. Bei der Anbahnung von Verträgen mit weiteren Kunden können somit Offerten des Wettbewerbs vorhergesehen und in die Verhandlungsstrategie eingeplant werden. Außerdem kann dieses Wissen ebenfalls für die Entscheidung über eine Rabattanfrage relevant sein. Prinzipiell ist also ein Wissenstransfer zwischen unterschiedlichen Geschäftsfällen notwendig. Insbesondere im Fall einer langwierigen Verhandlung gewinnt der Verkäufer Wissen über die Situation des Kunden und seine Marktkenntnisse. Darüber hinaus sammelt er Erfahrungen über seine Verhandlungsstrategie und bewertet deren Erfolg. Dieses Wissen kann situationsbedingt von ganz unterschiedlicher Relevanz sein und damit auf ganz unterschiedliche Art dokumentiert werden. Oftmals sollte das Wissen aber in späteren Geschäftsfällen zur Verfügung stehen, wobei zu beachten ist, dass der Verkäufer nicht das gesamte Know-how preisgeben möchte. Ein Teil des Wissens muss daher immer unter seiner persönlichen Kontrolle bleiben.

Der Verkäufer füllt nun die Rabattanfrage auf der Basis des Ergebnisses der Verhandlung mit dem Kunden aus. Dabei müssen die Anforderungen des Kunden möglichst auf vorgegebene Standardrabattierungen des Unternehmens umgesetzt werden. Falls dies nicht gelingt, erstellt der Verkäufer eine abweichende Rabattierung, die sich innerhalb gewisser Richtlinien bewegen muss, dem Verkäufer aber Spielräume offenhält. Bei der Erstellung einer Rabattanfrage kann es sich daher um eine komplexe Aufgabe handeln, bei der der Verkäufer - je nach einerseits Rabattierungsart, Sendungsaufkommen des Kunden, etc. und andererseits persönlicher Erfahrung - auf verschiedene Dokumente zurückgreifen muss. Darüber hinaus sichert er sich durch eine Rückfrage bei einem erfahrenen Kollegen ab, um keine fehlerhafte Rabattanfrage weiterzuleiten.

Das Informationssystem zur Erstellung von Rabattanfragen ist in einigen Bereichen kompliziert zu bedienen. Daher bestehen bei vielen Mitarbeitern Wissensbedarfe bzgl. der Bedienung des Systems, der richtigen Reaktion auf vom System angezeigte Meldungen, etc. Diese Wissensbedarfe sollten expliziert werden können, um einen entsprechenden zielgerichteten Wissensaufbau zu voranzutreiben.

Schließlich erstellt der Verkäufer die Rabattanfrage, wobei er seine Kompetenzen leicht überschreiten muss, um eine Bedingung des Kunden erfüllen zu können. Daher erstellt er eine Begründung, die er mit der Rabattanfrage an den Pricing Analyst weitergibt. In diesem Sonderfall entstehen also zusätzliche Ergebnisse.

Der Mitarbeiter im Preiskomitee erhält die Rabattanfrage und ist sich in diesem komplexen Fall unsicher, ob die Begründung des Verkäufers schlüssig ist. Für seine Entscheidungsfindung ist einerseits Wissen zu abgelaufenen Geschäftsfällen relevant, das von seiner Einschätzung der Rabattierung und des Verkäufers abhängt und damit nicht im Voraus planbar ist: Wurde zu diesem Kunden bereits vorher eine Vertragserstellung genehmigt oder abgelehnt? Welche

Rabattanfragen hat der genannte Verkäufer in letzter Zeit eingereicht, und konnten diese genehmigt werden? Andererseits greift er auf die Dokumente zu, die ihm seinen eigenen Entscheidungsspielraum verdeutlichen und Beispiele aufzeigen.

Abschließend kontaktiert er einen erfahrenen Kollegen, der ihm eine Begründung für die Genehmigung der Rabattanfrage liefert, die nicht in den offiziellen Dokumenten enthalten ist. Um die Gründe für seine Entscheidung für ihn und seine Vorgesetzten nachvollziehbar zu machen, müssen diese expliziert werden. Außerdem ist es wichtig nachzuhalten, wer in diesem Geschäftsfall als Experte fungiert hat. Auch dieses Wissen kann für die Bearbeitung späterer Geschäftsfälle relevant sein.

In diesem Abschnitt wurde exemplarisch die Bearbeitung eines einzelnen Geschäftsfalls insbesondere in Bezug auf das prozessbezogene Wissen dargestellt. Es sollte deutlich geworden sein, dass eine Unterstützung des Prozesses mit konventioneller Workflow-Management-Technologie nur unzureichend erfolgen kann. Bei den drei beschriebenen Aktivitäten „Rabattierung mit Kunden verhandeln“, „Rabattanfrage erstellen“ und „Rabattanfrage entscheiden“ handelt es sich um wissensintensive Aktivitäten gemäß der oben gegebenen Charakteristik. Der Prozess „Contract Management“ ist daher ein wissensintensiver Prozess, denn auch die weiteren in Abschnitt 4.1 geforderten Eigenschaften sind erfüllt: Der Prozess ist bis auf die erweiterten Anforderungen zum Management prozessbezogenen Wissens workflow-geeignet, er wird mehrmals pro Woche instanziiert, und die Durchlaufzeit liegt im Bereich weniger Tage. Der Prozess bildet mit den Prozessen zur Vertragsänderung und Vertragsauflösung eine Prozessgruppe. Die Prozesse dieser Gruppe haben einen starken inhaltlichen Bezug, da sie den Lebenszyklus eines Vertrags verwalten, und sie werden überwiegend von der gleichen Bearbeiterguppe, nämlich Verkäufer und Mitarbeiter der zentralen Abteilung Sales, durchgeführt.

C Konzeption einer Erweiterung von Workflow-Management-Systemen zur Unterstützung wissensintensiver Prozesse

- 5 Anforderungen an Workflow-Management-Systeme zur Unterstützung wissensintensiver Prozesse
- 6 Kontexte zur Unterstützung wissensintensiver Prozesse
- 7 Methodik zur Unterstützung wissensintensiver Prozesse durch Workflow-Management-Systeme

5 Anforderungen an Workflow-Management-Systeme zur Unterstützung wissensintensiver Prozesse

Im vorherigen Kapitel wurde die Klasse der wissensintensiven Prozesse eingeführt und motiviert, dass derzeitige WFMS diese Prozesse nur unzureichend ausführen können. Darauf aufbauend werden in diesem Kapitel Anforderungen an erweiterte WFMS zur Unterstützung wissensintensiver Prozesse vorgestellt. Der grundsätzliche Ansatz liegt dabei in der Erweiterung von WFMS um ein OMIS (Abschnitt 2.2.3), das den Anteil prozessbezogenen Wissens verwaltet, der von heutigen WFMS nicht unterstützt werden kann (im Weiteren als prozessbezogenes OMIS bezeichnet).

Der Begriff des prozessbezogenen Wissens ist der Ausgangspunkt der folgenden Überlegungen. Eine Workflow-Unterstützung von wissensintensiven Prozessen kann nicht ausschließlich auf den beschränkten Möglichkeiten heutiger WFMS zur Modellierung und Nutzung von Prozessmodell dokumenten beruhen. Daher soll zunächst diskutiert werden, welche Sichtweise auf prozessbezogenes Wissen dem in dieser Arbeit vorgestellten Ansatz zugrunde liegt (Abschnitt 5.1). Prozessbezogenes Wissen kann von ganz unterschiedlicher Charakteristik sein und unterschiedliche Aufgaben bei der Prozessbearbeitung übernehmen. Entsprechende Anforderungen werden in Abschnitt 5.2 vorgestellt. Um eine kontextabhängige Erfassung und Bereitstellung von Wissen anbieten zu können, ist insbesondere eine geeignete Strukturierung des Wissensbestands notwendig. Diese Strukturierung muss die Entstehung und Nutzung des Wissens im Kontext der aktuellen Situation des Bearbeiters bei der Ausführung von Prozessen durch WFMS berücksichtigen (Abschnitt 5.3). Ein Management prozessbezogenen Wissens sollte insbesondere Fragen des Aufbaus und der Bewahrung des Wissens einbeziehen. Aspekte, die den Wissensaufbau und die Wissensbewahrung in kooperativer Weise fördern, werden in Abschnitt 5.4 thematisiert. Weitere Anforderungen ergeben sich aus der Integration der Verwaltung prozessbezogenen Wissens in den Lebenszyklus von WFMS. Diese werden in Abschnitt 5.5 vorgestellt. Schließlich werden in Abschnitt 5.6 die aus den beschriebenen Anforderungen folgenden Teilaufgaben zur Konzeption einer Erweiterung von WFMS um eine Unterstützung wissensintensiver Prozesse skizziert.

5.1 Verwaltung prozessbezogenen Wissens

Verschiedene Ansätze zur Integration von Prozessmanagement und Wissensmanagement wurden in Kapitel 3 bereits vorgestellt. Dabei wurde deutlich, dass zwar in einer Vielzahl von Konzepten die Bedeutung des Managements prozessbezogenen Wissens hervorgehoben wird, aber keine einheitliche Sichtweise auf den Begriff des prozessbezogenen Wissens existiert.

Prozessbezogenes Wissen soll nach unserem Verständnis primär zu einer effektiven und effizienten Bearbeitung von Prozessen beitragen. Im Folgenden soll zunächst unterschieden werden, welche unterschiedlichen Quellen bei der Identifikation prozessbezogenen Wissens betrachtet werden müssen. Abbildung 25 gibt dazu einen Überblick über diese Wissensquellen und die zugehörigen Ausprägungen prozessbezogenen Wissens.

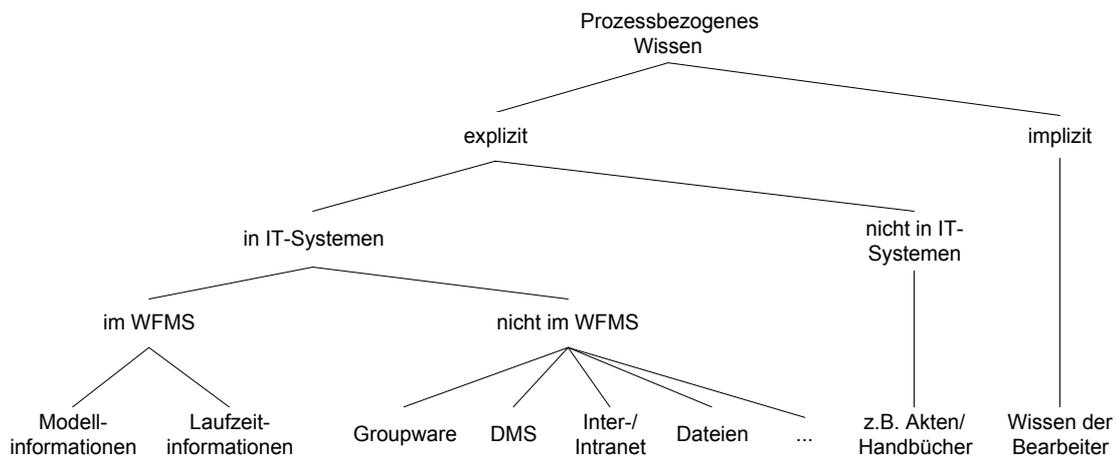


Abbildung 25: Unterschiedliche Quellen prozessbezogenen Wissens

Bei der Betrachtung prozessbezogenen Wissens kann grundsätzlich explizites und implizites Wissen unterschieden werden (Abschnitt 2.2.1). Der überwiegende Teil des expliziten prozessbezogenen Wissens wird in Unternehmen heutzutage in IT-Systemen verwaltet. Dabei sind für die vorliegende Arbeit WFMS, die sowohl Modellinformationen (z.B. Aktivitäten, deren Reihenfolge und die am Prozess beteiligten Rollen und Werkzeuge) als auch Laufzeitinformationen umfassen, von besonderer Bedeutung. Zu letzteren gehören Informationen über laufende und abgeschlossene Geschäftsfälle und insbesondere die Instanzen der Prozessmodell Dokumente. Ein erheblicher Teil prozessbezogenen Wissens wird außerhalb des WFMS in unterschiedlichen IT-Systemen verwaltet: Dazu gehören z.B. Groupware, Dokumenten-Management-Systeme, das Inter- bzw. Intranet oder das Filesystem. Explizites Wissen kann ferner auch nicht IT-gestützt vorliegen, hierzu zählen z.B. Akten oder in Papierform existierende Handbücher.

Der Schwerpunkt der Verwaltung prozessbezogenen Wissens soll in dieser Arbeit auf dem Bereich des expliziten Wissens liegen. Dennoch ist auch das implizite Wissen, also das Wissen, das bisher lediglich in den Köpfen der Mitarbeiter existiert, von eminenter Bedeutung. Eine geeignete Integration von Funktionalitäten eines prozessbezogenen OMIS in die WFMS-gestützte Bearbeitung von Prozessen zielt somit darauf ab, einen relevanten Teil des bisher impliziten Wissens zu explizieren und weiteren Mitarbeitern verfügbar zu machen. Es ist aber offensichtlich nicht sinnvoll und überhaupt möglich, das gesamte implizite Wissen der Mitarbeiter zu explizieren. Deshalb kann ein prozessbezogenes OMIS teilweise lediglich Verweise auf Wissensträger, wie z.B. Experten zur Bearbeitung einer Aktivität, enthalten, ohne dass das jeweilige Expertenwissen dokumentiert im OMIS vorliegen kann (vgl. Personifizierung und

Kodifizierung von Wissen in Abschnitt 2.2.3). Das Ziel beim Aufbau eines prozessbezogenen OMIS besteht somit darin, einen möglichst großen Teil des impliziten sowie expliziten prozessbezogenen Wissens zu integrieren und zu verwalten.

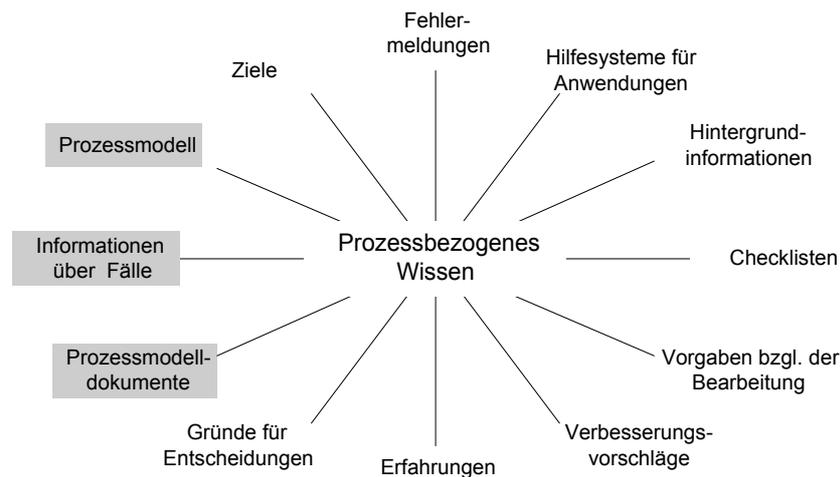


Abbildung 26: Beispiele für Typen prozessbezogenen Wissens

Abbildung 26 stellt einige relevante Beispiele für Typen prozessbezogenen Wissens dar (vgl. das Beispiel in Abschnitt 4.2 und eine ähnliche Darstellung in [Warg97]). Heutige WFMS unterstützen in vielen Fällen lediglich die grau hinterlegten Wissenstypen. Für wissensintensive Prozesse ist allerdings eine Menge weiterer Typen prozessbezogenen Wissens relevant, die mit Hilfe einer Erweiterung von WFMS durch ein prozessbezogenes OMIS geeignet unterstützt werden sollte.

5.2 Unterstützung unterschiedlicher Funktionen und Charakteristik prozessbezogenen Wissens

Wie im vorherigen Abschnitt dargelegt, soll prozessbezogenes Wissen zu einer effektiven und effizienten Bearbeitung von Prozessen beitragen. Die folgende Begriffsbildung zeigt in diesem Zusammenhang unterschiedliche Funktionen prozessbezogenen Wissens auf: Weber unterscheidet in [Webe99] primäre und sekundäre Informationsbedarfe. Dabei handelt es sich bei primären Informationsbedarfen um Informationsbedarfe, die sich direkt auf die Erfüllung einer Aufgabe beziehen, während bei sekundären Informationsbedarfen die Information einen unterstützenden Charakter hat. In Anlehnung an diese Begriffsbildung unterscheiden wir zwei Arten prozessbezogenen Wissens anhand der Begriffe „primäres Wissen“ und „sekundäres Wissen“ (in Bezug auf das in IT-Systemen repräsentierte Wissen werden im Folgenden entsprechend die Begriffe „primäres Dokument“ und „sekundäres Dokument“ verwandt).

Definition 13: Primäres Wissen

„Bei primärem Wissen handelt es sich um prozessbezogenes Wissen, das durch das zugehörige Prozessmodell festgelegt ist.“

Primäres Wissen ist prinzipiell für jeden zu bearbeitenden Geschäftsfall relevant und erforderlich. Zu primärem Wissen zählen einerseits die im Prozessmodell auf Typebene spezifizierten Informationen als auch die entsprechenden Informationen auf Instanz- bzw. Geschäftsfallebene. Somit verstehen wir unter primärem Wissen sowohl im Prozessmodell repräsentierte Informationen über z.B. die Aktivitäten eines Prozesses, die jeweils zu beteiligenden Bearbeiter und die Prozessmodelldokumente als auch die entsprechenden Informationen für jeden einzelnen Geschäftsfall. Nach Definition 8 ist das primäre Wissen somit für ein WFMS ausreichend, um Prozesse ausführen zu können.

Definition 14: Sekundäres Wissen

„Bei sekundärem Wissen handelt es sich um prozessbezogenes Wissen, das erstens zwar zum Modellierungszeitpunkt prinzipiell bekannt ist, aber dessen Relevanz sich erst situativ im Kontext einer konkreten Geschäftsfallbearbeitung ergibt oder das zweitens zum Zeitpunkt der Modellierung noch nicht bekannt ist, da das Wissen erst während der Prozessdurchführung gewonnen wird.“

Im Gegensatz zu primärem Wissen kann sekundäres Wissen somit nicht sinnvollerweise in einem Prozessmodell festgelegt werden. Vielmehr muss sekundäres Wissen durch ein prozessbezogenes OMIS verwaltet werden, das geeignet in ein WFMS zu integrieren ist. Da die Konzeption eines prozessbezogenen OMIS den Schwerpunkt dieser Arbeit darstellt, werden wir im Folgenden Anforderungen aufführen, die die Verwaltung, Strukturierung und Bewahrung von sekundärem Wissen zum Inhalt haben.

Tabelle 2 veranschaulicht die beiden Begriffe anhand des Beispiels „Contract Management“. Sowohl primäres als auch sekundäres Wissen ist für die effektive und effiziente Bearbeitung eines wissensintensiven Prozesses notwendig.

Sekundäres Wissen besitzt verschiedene Eigenschaften, die in einem prozessbezogenen OMIS geeignet abgebildet werden müssen, um dem Bearbeiter eine optimale Unterstützung bei der Durchführung von Prozessen zu gewähren. Aspekte, die hier insbesondere betrachtet werden müssen, sind Fragen, wann das Wissen benötigt wird, ob das Wissen nur unter bestimmten Umständen, also z.B. in Abhängigkeit von Eigenschaften des Geschäftsfalls von Interesse ist, oder ob das Wissen zu bestimmten Ereignissen der Prozessdurchführung aktiv angeboten werden soll. Darüber hinaus muss für Bearbeiter die Möglichkeit bestehen, den Zugriff weiterer Benutzer auf bestimmte Wissensbestände einzuschränken, um z.B. Erfahrungen oder eigenes Know-how unter eigener Kontrolle zu halten. Die diesbezüglich relevanten Eigenschaften sekundären Wissens sind zu untersuchen, in geeignete Funktionalitäten eines prozessbezogenen OMIS umzusetzen und in ein WFMS zu integrieren.

Tabelle 2: Beispiele für primäres und sekundäres Wissen

Primäres Wissen	Sekundäres Wissen
Aktivitäten Bearbeitungsreihenfolge von Aktivitäten Bearbeiter Werkzeuge Prozessmodelldokumente: - Rabattanfrage - Simulation der Rabattanfrage - Rabattbescheid - Vertrag und Anlagen	Vorgabe zu Rabattierungen zu Key Accounts Angebote der Wettbewerber Marktkennnisse des Kunden Erfahrungen bzgl. der Verhandlungsstrategie Expertenwissen zur Erstellung einer Rabattanfrage Fehlermeldungen FAQ Begründung einer Entscheidung Vorgabe zu Entscheidungsspielräumen Informationen über ähnliche Geschäftsfälle ...

Eine Kernanforderung besteht diesbezüglich darin, möglichst viel an relevanten Informationen aus dem WFMS zur Beschreibung der Eigenschaften sekundären Wissens zu nutzen. Dies kann in ganz unterschiedlichen Situationen sinnvoll sein. Betrachten wir dazu wiederum das Beispiel „Contract Management“: Eine Änderung der Richtlinie zu möglichen Rabattierungen bei Großkunden soll dem Verkäufer zum richtigen Zeitpunkt, also bei der Vorbereitung auf ein Kundengespräch, zugeleitet werden. Dieser Zeitpunkt kann offensichtlich durch das WFMS bestimmt werden, da es sich bei dem Kundengespräch um eine Aktivität im Prozess handelt, über deren Aktivierung durch den Verkäufer das WFMS benachrichtigt wird. Das WFMS kann somit die entsprechenden Aktionen zur Zustellung des Dokuments einleiten. Ein weiteres Beispiel besteht darin, dass sekundäre Dokumente oftmals nur unter bestimmten Umständen relevant sind, also wenn z.B. die Frist zur Bearbeitung einer Aktivität überschritten ist, der zugehörige Geschäftsfall eine hohe Priorität besitzt oder Ausnahmen aufgetreten sind. Das Eintreten dieser Bedingungen kann aber zum überwiegenden Teil aus Informationen des WFMS abgeleitet werden. Somit ist auch in diesem Fall eine möglichst starke Kopplung der Beschreibung sekundären Wissens an Informationen aus dem WFMS sinnvoll.

5.3 Strukturierung sekundären Wissens anhand des Bearbeitungskontextes

Die angemessene Strukturierung und dementsprechende Präsentation sekundären Wissens stellt eine der Kernanforderungen für die vorliegende Arbeit dar. Wie in den vorherigen Abschnitten deutlich wurde, steht eine große Menge von Wissen in einem Bezug zu der Bearbeitung von Prozessen. Dabei sind diese Wissensbestände aber im Allgemeinen lediglich in einem bestimmten Bearbeitungskontext relevant. Dieser Bearbeitungskontext kann z.B. durch die Durchführung einer bestimmten Aktivität oder durch die Nutzung eines bestimmten Werk-

zeugs charakterisiert sein. Es ist also nicht ausreichend, wenn ein prozessbezogenes OMIS die in Abschnitt 5.1 aufgeführten Wissenstypen speichert und dem Bearbeiter das Auffinden der für ihn jeweils notwendigen Informationen überlässt. Vielmehr ist eine kontextorientierte Unterstützung gefordert (vgl. „contextual assistance“ in [Nutt96]), die dem Bearbeiter durch eine geeignete Strukturierung die jeweils relevanten Informationen aus dem prozessbezogenem OMIS präsentiert.

Um den Zugang zu dem in einem prozessbezogenen OMIS repräsentierten Wissen zu erleichtern, schlagen wir die Auswahl eines primären Strukturierungsmerkmals vor, das die Perspektive auf das Wissen ordnet. Nicht immer ist es sinnvoll, mit der Auswahl eines vorrangigen Merkmals eine verbindliche Ordnungsstruktur vorzugeben. Insbesondere ist es wichtig, dass den Nutzern einer Ordnungsstruktur die einzelnen Ordnungsmerkmale bekannt sind. Im Fall des prozessbezogenen Wissens finden wir mit den Objekten des Workflow-Metamodells, wie Prozess, Aktivität, Organisationseinheit oder Werkzeug (Abschnitt 2.1.1), allgemeine den Bearbeitern bekannte Ordnungsmerkmale vor. Die Bearbeiter kennen diese Ordnungsmerkmale, da sie gemäß dem partizipativen Ansatz zur Workflow-Management-Einführung (Abschnitt 2.1.2) von Anfang an bei der Erhebung und Modellierung beteiligt wurden und somit nicht nur bzgl. ihrer eigenen Aufgaben kompetent sind, sondern sich auch im gesamten Prozess zurechtfinden. Die Klassifikation sekundären Wissens gemäß der beschriebenen Ordnungsstruktur muss nun so ausgestaltet werden, dass die Anzahl der Ordnungsmerkmale einerseits groß genug ist, um überhaupt eine echte Strukturierung zu gewährleisten, und andererseits für den Bearbeiter noch überschaubar ist. Dies ist ferner eine wichtige Voraussetzung für eine erfolgreiche Bewahrung sekundären Wissens (siehe nächster Abschnitt).

In Bezug auf das primäre Strukturierungsmerkmal unterscheidet sich ein prozessbezogenes OMIS von den meisten OMIS-Ansätzen. Da ein klassisches OMIS die Verwaltung des gesamten oder zumindest eines relevanten Anteils des Unternehmenswissens zum Ziel hat, dieses Wissen aber nicht notwendigerweise in einen Bezug zu definierten oder sogar durch WFMS unterstützten Prozessen gesetzt wird, bieten diese Systeme im Allgemeinen keine Strukturierung des Wissens anhand von Prozessen, Aktivitäten, etc. an. Vielmehr sollen durch Funktionalitäten zur inhaltlichen Suche im gespeicherten Dokumentenbestand die Informationsbedarfe der Benutzer erfüllt werden. In Bezug auf die in Abschnitt 2.3 vorgestellten Ansätze zur Dokumentenmodellierung steht also bei klassischen OMIS die semantische Sicht im Vergleich zur Kontextsicht klar im Vordergrund.

Prozessbezogene OMIS dagegen verwalten Informationen, die für die Durchführung von Prozessen relevantes Wissen repräsentieren. In den meisten Fällen sind die Bearbeiter in der Lage zu bestimmen, in welcher Bearbeitungssituation das entsprechende Wissen relevant ist, und damit eine Verknüpfung mit Workflow-Objekten, wie z.B. einer bestimmten Aktivität, festzulegen. Inhaltsorientierte Suchmechanismen sind zwar für ein prozessbezogenes OMIS ebenfalls wünschenswert, die geforderte kontextorientierte Unterstützung wird aber maßgeblich durch das skizzierte Konzept zur Strukturierung sekundären Wissens bereitgestellt. Bei prozessbezogenen OMIS steht somit die Kontextsicht für die Abbildung der relevanten Aspekte von Dokumenten im Vordergrund, während die semantische Sicht (und entsprechende aufwändige semantische Ansätze) hier nicht die Bedeutung besitzt wie bei klassischen OMIS.

Ein wesentlicher Vorteil dieses Ansatzes besteht darin, dass die Informationen zum Aufbau des Strukturierungsrahmens, nämlich die Workflow-Objekte, direkt vom WFMS bereitgestellt werden können. Damit entfallen zeitaufwändige und komplexe Aktivitäten zur zusätzlichen Erstellung und Wartung einer Ordnungsstruktur, da das WFMS stets die aktuellen Prozess- und Organisationsstrukturen enthält. Wir sehen daher die Kopplung von WFMS und prozessbezogenen OMIS zur Strukturierung sekundären Wissens anhand des Bearbeitungskontexts als ein Kernkonzept an, um eine Integration von Workflow Management und Wissensmanagement zu ermöglichen.

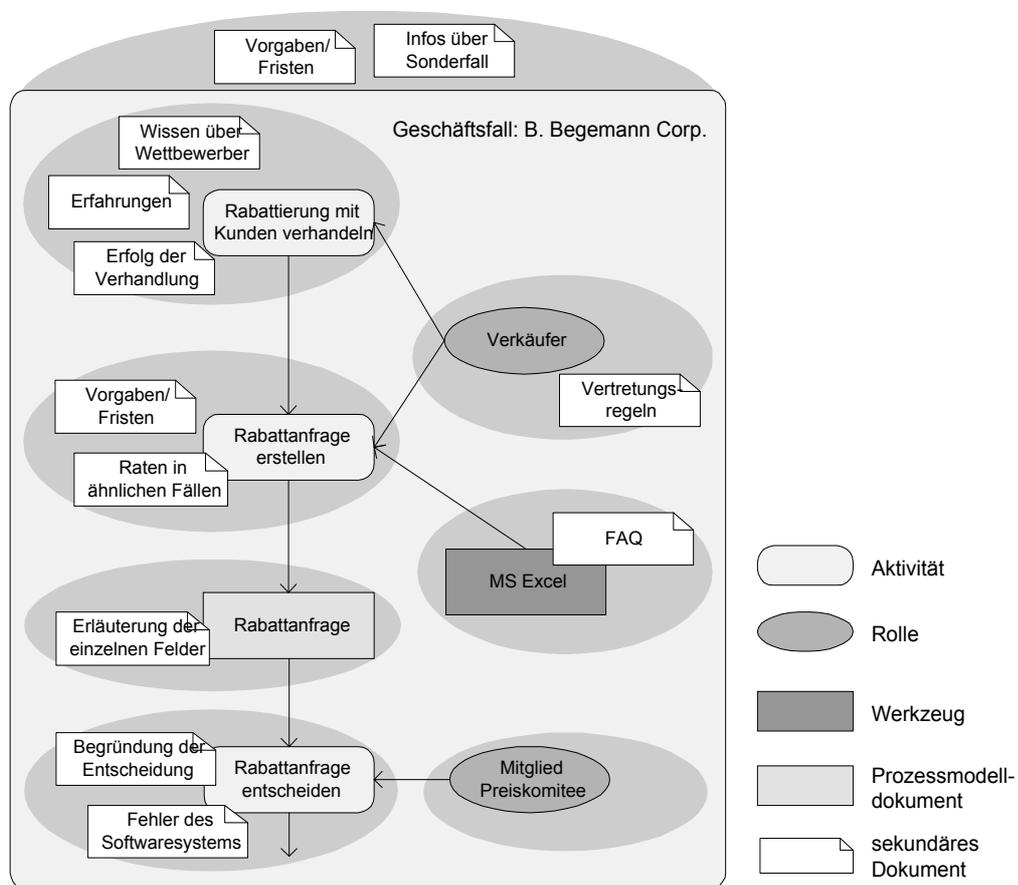


Abbildung 27: Strukturierung sekundären Wissens anhand der Workflow-Objekte

Im Folgenden wollen wir die oben skizzierte Strukturierung beispielhaft erläutern und motivieren, wie auf diese Weise eine angemessene Unterstützung wissensintensiver Prozesse ermöglicht wird. Abbildung 27 zeigt dazu anhand eines Ausschnitts aus dem Prozess „Contract Management“ exemplarisch die Zuordnung von Wissensbeständen zu Workflow-Objekten, um die beschriebene Strukturierung sekundären Wissens zu illustrieren: Bei der Verhandlung mit dem Kunden benötigt oder erstellt der Verkäufer verschiedene Dokumente („Wissen über Wettbewerb“, „Erfahrungen“, „Erfolg der Verhandlung“), die einerseits nur für diese Aktivität relevant sind, andererseits aber unabhängig von konkreten Geschäftsfällen sind. Durch eine

solche Zuordnung von Wissen zu Aktivitäten kann z.B. ein Wissenstransfer zwischen unterschiedlichen Geschäftsfällen unterstützt werden, wie er bei wissensintensiven Prozessen verstärkt auftritt (Abbildung 22). Weitere Wissensbestände sind bei der Benutzung von Werkzeugen von Interesse („FAQ“). Da Werkzeuge aber oftmals in unterschiedlichen Aktivitäten innerhalb eines Prozesses benutzt werden, sind diese Wissensbestände unabhängig von der Nutzung in konkreten Aktivitäten, sondern sie sind allgemein beim Einsatz des Werkzeugs relevant. Darüber hinaus sind weitere Strukturierungsmerkmale sinnvoll und notwendig: So ist z.B. bestimmtes Wissen bei der Durchführung sämtlicher Aktivitäten relevant, aber auf einen spezifischen Geschäftsfall beschränkt („Vorgaben/Fristen“, „Infos über Sonderfall“), wodurch der Wissenstransfer innerhalb von Geschäftsfällen verbessert werden kann. Während der Geschäftsbearbeitung werden dem Bearbeiter vom WFMS die entsprechenden Prozessmodelldokumente gemäß des Prozessmodells zugestellt (der Prozess wird also durch die klassische Workflow-Funktionalität gesteuert), und darüber hinaus erhält er über die Funktionalitäten des prozessbezogenen OMIS Zugriff auf die sekundären Dokumente, die für seine augenblickliche Bearbeitungssituation von besonderer Bedeutung sind.

Das WFMS kann aber nicht nur den grundsätzlichen Strukturierungsrahmen für sekundäres Wissen festlegen. Zusätzlich kann aus den Informationen des WFMS der aktuelle Bearbeitungskontext für einen Bearbeiter erschlossen werden und bei einer Ablage und Nutzung von Wissen genutzt werden. Da das WFMS den Bearbeitungskontext kennt, also die Information liefern kann, welche Aktivität in welchem Geschäftsfall ggf. mit welchem Werkzeug, etc. der Bearbeiter gerade ausführt, kann ein prozessbezogenes OMIS dem Bearbeiter die Menge der potentiell für ihn relevanten Dokumente geeignet einschränken und präsentieren.

5.4 Kooperativer Aufbau und Bewahrung sekundären Wissens

In den vorherigen Abschnitten wurde erläutert, welche Sichtweise auf prozessbezogenes und insbesondere sekundäres Wissen dem Ansatz dieser Arbeit zugrunde liegt und wie eine geeignete Strukturierung sekundären Wissens eine Unterstützung für Bearbeiter bieten kann. Im Folgenden soll nun darauf aufbauend der Frage nachgegangen werden, wie ein prozessbezogenes OMIS den Aufbau, die Verteilung und die Bewahrung von sekundärem Wissen bei der Durchführung wissensintensiver Prozesse sicherstellen kann.

Wie in Kapitel 3 dargestellt, können bei der Analyse vorhandener Ansätze zur Integration von Workflow Management und Wissensmanagement zwei Konzepte bzgl. der Operationen, die dem Bearbeiter auf dem verwalteten Wissensbestand zur Verfügung stehen, unterschieden werden:

Ein Teil der betrachteten Systeme beschränkt sich in Bezug auf den Wissensaufbau und die Wissensbewahrung auf die Spezifikation der Wissensbedarfe der Bearbeiter (bspw. zur Bearbeitung einer bestimmten Aktivität) zum Modellierungszeitpunkt. Die Wissensbedarfe werden durch die Spezifikation von Agenten oder kontextabhängigen Sichten repräsentiert, die zur Laufzeit mit den Eigenschaften der konkreten Bearbeitungssituation parametrisiert dem Bearbeiter eine Menge von entsprechenden Dokumenten aus dem verwalteten Wissensbestand lie-

fern. Der Bearbeiter hat bei diesen Systemen keine direkte Möglichkeit, integriert in die Prozessbearbeitung eigenes Wissen im System abzulegen oder veraltetes Wissen zu entfernen, um zu einem Wissensaufbau bzw. einer Wissensverteilung und Wissensbewahrung beizutragen. Seine Rolle beschränkt sich auf die Nutzung von Wissen ohne direkte Einflussnahme auf den Wissensbestand. Insbesondere wenn gut formalisierbares Faktenwissen verwaltet werden soll, wie z.B. rechtliche Vorgaben, kommen die in Abschnitt 2.3 skizzierten Möglichkeiten zur inhaltlichen Repräsentation von Dokumenten zum Einsatz. Bei den Systemen, die diesen Ansatz in ausgeprägter Weise umsetzen, handelt es sich um KnowMore, Ontobroker und EULE2.

Der zweite Ansatz stellt den Bearbeiter mehr in die Rolle des Wissensmanagers. Zwar werden auch hier bestimmte Wissensbestände bereits zum Zeitpunkt der Modellierung als relevant für bestimmte Bearbeitungssituationen vorgegeben, diese sind aber zur Laufzeit nicht unveränderlich: Der Bearbeiter kann die vordefinierten Wissensbestände bei der Durchführung von Prozessen nutzen und darüber hinaus integriert in die Prozessbearbeitung neues Wissen in Form von Dokumenten in das System einfügen bzw. vorhandene Dokumente modifizieren oder aus dem System entfernen. Der Bearbeiter hat also selbst die Möglichkeit, zu einem kooperativen Aufbau und einer kooperativen Bewahrung von Wissen beizutragen. Lediglich die Systeme MILOS und WorkBrain zeigen Ansätze in diese Richtung.

Wir halten den zweiten Ansatz für die Unterstützung wissensintensiver Prozesse aus folgenden Gründen für erfolgversprechender: Wie im vorherigen Kapitel dargestellt, zeichnen sich wissensintensive Prozesse durch eine hohe Flexibilität in Bezug auf sowohl die Wissensbedarfe als auch die Ergebnisse ihrer Kernaktivitäten, den wissensintensiven Aktivitäten, aus. Zusätzlich sind wissensintensive Prozesse durch einen hohen Wissenstransfer innerhalb von Geschäftsfällen und zwischen Geschäftsfällen charakterisiert. Dies impliziert, dass der überwiegende Teil des zur Durchführung relevanten Wissens nicht bereits zum Modellierungszeitpunkt vorgegeben werden kann. Vielmehr muss der einzelne Bearbeiter - unterstützt durch organisatorische Regelungen und entsprechende Funktionalitäten eines erweiterten WFMS - die Möglichkeit besitzen, sowohl vorhandenes sekundäres Wissen zu nutzen als auch neues sekundäres Wissen in Form von Dokumenten zu veröffentlichen und den Wissensbestand zu aktualisieren. Entsprechende Funktionalitäten sind durch das prozessbezogene OMIS aufbauend auf der im vorherigen Abschnitt erläuterten Strukturierung prozessbezogenen Wissens und integriert in die Workflow-Bearbeitung bereitzustellen.

Es sind ferner organisatorische Regelungen notwendig, die aufzeigen, welche Personen für die Bewahrung der einzelnen sekundären Wissensbestände verantwortlich sind. Insbesondere sind organisatorische Rollen mit unterschiedlichen Verantwortungsbereichen in Bezug auf die Bewahrung sekundären Wissens zu definieren. Die Strukturierung des Wissens anhand des Bearbeitungskontextes soll hierfür als eine Basis für die Erstellung dieses organisatorischen Regelwerks dienen. Die geforderten Organisationskonzepte zur Bewahrung sekundären Wissens sind einzubetten in den Prozesslebenszyklus und damit in die Aufgaben, die bereits bei heutigen WFMS von verschiedenen Rollen zur Bewahrung von Prozessen und prozessbezogenen Informationen durchgeführt werden. Informationstechnische Maßnahmen, wie wissensbezogene Attribute zu Dokumenten (zu diesen gehören die Angabe von

Gültigkeitszeiträumen oder weitere Attribute, die den Lebenszyklus von Dokumenten beschreiben) sollen dem Bearbeiter zusätzliche Unterstützung bieten. Mit Hilfe dieser Konzepte ist das Problem der Wissensbewahrung zu bewältigen.

5.5 Integration der Verwaltung sekundären Wissens in den Prozesslebenszyklus

In Abschnitt 2.1.2 sind die einzelnen Phasen des Prozesslebenszyklus und die unterschiedlichen beteiligten Rollen kurz vorgestellt. Aus den bisher aufgeführten Anforderungen folgt nun, dass das Management sekundären Wissens geeignet in die Erstellung und kontinuierliche Verbesserung einer Workflow-Management-Anwendung zu integrieren ist. Damit ergibt sich als neue Anforderung, dass eine Erweiterung von WFMS um ein prozessbezogenes OMIS in der Form erfolgen muss und um ein entsprechendes organisatorisches Konzept bereichert werden muss, sodass die hierfür relevanten vorgestellten Bausteine des Wissensmanagements (Abschnitt 2.2.1) in den Prozesslebenszyklus integriert werden. Die grundsätzliche Sicht auf eine Integration von Wissensbausteinen in den Prozesslebenszyklus soll im Folgenden vorgestellt werden.

Die Phase der Erhebung und anschließender Modellierung des Prozesses beinhaltet die Identifikation prozessbezogenen Wissens. Bei der klassischen Vorgehensweise zur Erstellung einer Workflow-Management-Anwendung wird in diesen Phasen lediglich das primäre Wissen erhoben und durch das Prozessmodell repräsentiert. Für eine Unterstützung wissensintensiver Prozesse soll zusätzlich eine Identifikation des sekundären Wissens erfolgen, soweit dies zum Modellierungszeitpunkt überhaupt bekannt ist. Dazu muss ein erweitertes WFMS Möglichkeiten bereitstellen, die dem Modellierer erlauben, das identifizierte sekundäre Wissen in ein prozessbezogenes OMIS einzustellen und gemäß des in Abschnitt 5.3 diskutierten Ordnungskonzepts zu strukturieren, um so einen initialen Wissensbestand festlegen zu können.

Wie bereits in Abschnitt 5.4 erläutert, stellt die Anwendung und flexible Anpassung der Workflow-Management-Anwendung die zentrale Phase bei der Betrachtung prozessbezogenen Wissens dar, da der Aufbau (abgesehen von dem bereits während der Modellierung erstellten Wissensbestand) und insbesondere die Bewahrung des Wissens kooperativ durch die Bearbeiter während der Durchführung von Prozessen erfolgen soll. Daher sind die Bausteine Wissensentwicklung, Wissensverteilung, Wissensnutzung und Wissensbewahrung diesen beiden Phasen des Prozesslebenszyklus zugeordnet. Diesbezügliche Anforderungen, wie die integrierte Nutzung von sekundärem Wissen während der Workflow-Bearbeitung, die ggf. aktive Benachrichtigung von Bearbeitern über relevante Informationen durch das System oder Fragen der Wissensbewahrung wurden bereits in den vorherigen Abschnitten genannt.

Wissenserwerb, also das externe Einkaufen von Wissen, spielt dagegen für wissensintensive Prozesse im hier definierten Sinn keine Rolle und kann daher vernachlässigt werden. Schließlich muss mit der Evaluation der Workflow-Management-Anwendung eine Bewertung des prozessbezogenen Wissens durchgeführt werden, um den Regelkreis aus Zielfestlegung, Umsetzung und Evaluation schließen und damit Feedback für eine erneute Festlegung von Prozess- und Wissenszielen zu liefern.

Eine Abgrenzung der Arbeit ergibt sich aus der Forderung, dass prozessbezogenes Wissen direkt die Durchführung von wissensintensiven Prozessen verbessern soll. Betrachtet man den gesamten Workflow-Lebenszyklus, wird bei der Erhebung von Prozessen Wissen gewonnen, das für die initiale Modellierung des Prozesses wichtig ist. Es handelt sich dabei um Interviewprotokolle, schriftliche Befragungen von Mitarbeitern, Ergebnisse der Dokumentenanalyse, in Form von Videos dokumentierten Beobachtungen, etc. (das Konzept eines OMIS zur Unterstützung der Erhebung und Modellierung wird in [Kuhl99] beschrieben). Ein Teil dieses Wissens kann nach einer entsprechenden Aufbereitung ebenfalls als sekundäres Wissen für die Durchführung des Prozesses Hilfestellung bieten. Wir werden in dieser Arbeit eine Strukturierung sekundären Wissens erarbeiten, die dem Modellierer erlaubt, dieses Wissen entsprechend der Relevanz für die konkrete Bearbeitungssituation bereitzustellen. Der Prozess der Auswahl des bei der Prozesserhebung gewonnenen Wissens und die Aufgaben der zielgruppengerechten Aufbereitung dieses Wissens werden in dieser Arbeit nicht betrachtet.

5.6 Teilaufgaben bei der Entwicklung eines Konzepts zur Unterstützung wissensintensiver Prozesse durch Workflow-Management-Systeme

Ziel dieser Arbeit ist die Erweiterung eines WFMS um ein prozessbezogenes OMIS zur Unterstützung wissensintensiver Prozesse. Im Folgenden ist ein konzeptioneller Rahmen zu entwickeln, der die dargestellten Anforderungen erfüllt und umsetzt. Dieser konzeptionelle Rahmen orientiert sich am Begriff des „Kontexts“. Ein Kontext enthält eine Menge von Dokumenten, die den Bearbeiter bei der workflow-gestützten Durchführung von Prozessen unterstützen sollen, wobei jeder Kontext zur Unterstützung einer konkreten Bearbeitungs- bzw. Nutzungssituation dient (eine genauere Definition erfolgt im nächsten Kapitel). Die konzeptionelle Ausarbeitung von Kontexten wird in den folgenden Schritten durchgeführt:

- Erarbeitung eines Strukturierungsrahmens und damit einer Menge von praxisrelevanten Typen von Kontexten (Abschnitt 6.2)
- Konzeption eines Informationsmodells für in Kontexten verwaltete Dokumente, das den unterschiedlichen Funktionen und der unterschiedlichen Charakteristik sekundären Wissens Rechnung trägt (Abschnitt 6.3)
- Organisatorische Einbettung der Verwaltung und der Nutzung von Kontexten in den Prozesslebenszyklus (Kapitel 7)
- Erweiterung von WFMS zur informationstechnischen Unterstützung von Kontexten (Kapitel 8)
- Prototypische Realisierung von Kernaspekten des Konzepts (Kapitel 9)

Die einzelnen Aspekte werden im Folgenden genauer erläutert.

Kontexte sollen die Strukturierung des Wissens anhand des Bearbeitungskontexts erlauben und dem Bearbeiter dadurch das jeweils relevante sekundäre Wissen anbieten (Abschnitt 5.3). Sie setzen dazu das sekundäre Wissen in Beziehung zu Workflow-Objekten, d.h. den vom WFMS verwalteten Objekten, wie Aktivität „Rabattanfrage erstellen“ oder Geschäftsfall

„comdirect bank“. Nicht jedes Workflow-Objekt ist aber sinnvollerweise kontextbildend, d.h. es dient zur Strukturierung sekundären Wissens für eine bestimmte Bearbeitungs- bzw. Nutzungssituation. So können zwar offensichtlich Aktivitäten als kontextbildende Objekte angesehen werden, indem die entsprechenden Kontexte jeweils die Dokumente enthalten, die die Bearbeitung der jeweiligen Aktivität unterstützen, bei anderen vom WFMS verwalteten Objekten, wie Zugriffsrechten oder WFMS-internen Steuerungsdaten, handelt es sich aber sinnvollerweise nicht um kontextbildende Objekte. Daher wird in Abschnitt 6.2 eine Strukturierung sekundären Wissens dargestellt, indem aufbauend auf einer Analyse existierender Metamodelle des Workflow Managements praxisrelevante kontextbildende Objekttypen identifiziert werden. Ziel ist es, dem Bearbeiter eine Ordnungsstruktur zur Verwaltung sekundären Wissens zur Verfügung zu stellen, die ihm bekannt ist und die ihm die Suche nach relevantem Wissen für seine aktuelle Bearbeitungssituation erleichtert. Die Ordnungsstruktur soll automatisch durch das WFMS aufgebaut werden können, da die entsprechenden kontextbildenden Objekte vom WFMS verwaltet werden. Zusätzlich wird die Menge und die Struktur der aus den kontextbildenden Objekten abgeleiteten Typen von Kontexten erläutert und jeweils anhand von Verwendungsregeln für den Bearbeiter und Beispielen gezeigt, welchen Zweck ein Kontext für die Bearbeitung wissensintensiver Prozesse erfüllt.

Neben einem Strukturierungsrahmen für Kontexte ist eine Beschreibung der Dokumente in Kontexten notwendig, die den Anforderungen der Verwaltung sekundären Wissens verschiedener Quellen und Typen (Abschnitt 5.1) und der unterschiedlichen Funktion und Charakteristik sekundären Wissens (Abschnitt 5.2) Rechnung trägt. Daher präsentieren wir in Abschnitt 6.3 ein Informationsmodell für Kontextdokumente, dessen einzelne Aspekte die Umsetzung der genannten Anforderungen sicherstellen. Kernelemente dieses Informationsmodells sind Lebenszyklusaspekte von Dokumenten, die Spezifikation von Sichtbarkeitsbedingungen für Dokumente in Abhängigkeit von Eigenschaften des aktuell vom Bearbeiter durchgeführten Geschäftsfalls und die aktive Bereitstellung von Dokumenten in Abhängigkeit von Ereignissen der Geschäftsfallbearbeitung.

Die zentrale Anforderung in Bezug auf die organisatorische Einbettung von Kontexten besteht in der geeigneten Integration sekundären Wissens und damit des Kontextkonzepts in den Prozesslebenszyklus (Abschnitt 5.5). In Kapitel 7 wird daher dargestellt, welche Aufgaben unterschiedliche Gruppen bzw. Rollen, wie z.B. Modellierer, Kontextverantwortliche oder Bearbeiter, in den einzelnen Phasen des Prozesslebenszyklus in Bezug auf die Verwaltung von Kontexten wahrnehmen. Dabei liegt der Schwerpunkt der Betrachtung auf der Fragestellung, mit welchen insbesondere organisatorischen Mitteln die kooperative Erstellung und Bewahrung von sekundärem Wissen sichergestellt werden kann (Abschnitt 5.4).

Für eine informationstechnische Umsetzung des Kontextkonzepts sind WFMS in verschiedener Hinsicht zu erweitern. Wir werden daher in Kapitel 8 sowohl eine Systemarchitektur für in diesem Sinne erweiterte WFMS vorstellen als auch jeweils den Beitrag der einzelnen Komponenten zur Verwaltung prozessbezogenen Wissens erläutern. Dabei stellen wir insbesondere einen Vorschlag zur Präsentation von Kontexten innerhalb des Workflow-Clients vor.

6 Kontexte zur Unterstützung wissensintensiver Prozesse

Wie wir im vorherigen Kapitel dargestellt haben, sind WFMS in verschiedener Hinsicht für eine angemessene Unterstützung wissensintensiver Prozesse zu erweitern. Die Einbeziehung sekundären Wissens, die Möglichkeit zum kooperativen Aufbau und zur Bewahrung sekundären Wissens und die aktive Präsentation relevanter Informationen, unterstützt durch eine entsprechende Strukturierung, stellen die Kernanforderungen dar, die in diesem Kapitel durch ein entsprechendes Konzept umgesetzt werden sollen. Dazu werden zunächst einige Begriffe eingeführt und der Aufbau und die Funktion der folgenden Abschnitte in diesem Kapitel erläutert.

6.1 Überblick

Als konzeptionellen Rahmen für die weiteren Überlegungen führen wir den Begriff des „Kontexts“ ein, den wir in diesem Kapitel schrittweise entwickeln werden (die folgenden Überlegungen basieren auf grundlegenden Ideen zu Kontexten, die wir bereits in [DGL00] und [GH00] vorgestellt haben). Dazu sind zunächst einige Definitionen notwendig.

Definition 15: Workflow-Objekt und Workflow-Objekttyp

„Bei einem Workflow-Objekt handelt es sich um ein vom WFMS verwaltetes Objekt, das entweder Teil einer Prozess- oder Organisationsdefinition ist oder einen Geschäftsfall repräsentiert. Die entsprechenden Typen von Workflow-Objekten bezeichnen wir als Workflow-Objekttypen.“

Beispiele für Workflow-Objekte sind „Rabattanfrage erstellen“ und „Rabattanfrage entscheiden“ des Workflow-Objekttyps „Aktivität“ oder „Vertrag Fa. Busch“ des Workflow-Objekttyps „Geschäftsfall“. Die Einschränkung der Workflow-Objekte auf den Bereich der Prozess- und Organisationsdefinition wird Abschnitt 6.2 rechtfertigen.

Definition 16: Kontextbildendes Objekt und kontextbildender Objekttyp

„Ein kontextbildendes Objekt ist ein Workflow-Objekt, das zur Bildung eines Kontexts oder mehrerer Kontexte und damit zur Strukturierung sekundären Wissens dient. Die entsprechenden Typen von kontextbildenden Objekten bezeichnen wir als kontextbildende Objekttypen.“

Wir haben bereits im letzten Kapitel dargestellt, dass Workflow-Objekte wie „Prozess“, „Aktivität“ oder „Geschäftsfall“ zur Strukturierung sekundären Wissens geeignet sind und somit kontextbildende Objekte darstellen. Dies ist selbstverständlich nicht für alle vom WFMS verwalteten Workflow-Objekte der Fall. Daher besteht eine der zentralen konzeptionellen Aufgaben in der Identifikation einer relevanten und praxistauglichen Menge von kontextbildenden Objekten bei WFMS.

Definition 17: Kontext und Kontexttyp

„Ein Kontext besteht aus einer Menge von Kontextdokumenten, und er ist an ein kontextbildendes Objekt oder an mehrere kontextbildenden Objekte gebunden.“

Die Dokumente, die ein Kontext enthält, sollen den Bearbeiter in einer konkreten Bearbeitungssituation bei der workflow-gestützten Durchführung von Prozessen unterstützen. Die Bearbeitungssituation wird durch die Menge der kontextbildenden Objekte, an die der Kontext gebunden ist, vorgegeben. Ist also beispielsweise ein Kontext an das kontextbildende Objekt „Rabattanfrage erstellen“ (vom Objekttyp „Aktivität“) gebunden, dann repräsentieren die im Kontext enthaltenen Dokumente relevantes Wissen für die Durchführung dieser konkreten Aktivität.

„Ein Kontexttyp ist durch die Bindung an einen oder mehrere kontextbildende Objekttypen festgelegt.“

Kontexttypen enthalten keine Dokumente. Es handelt sich bei Kontexttypen um Hilfskonstrukte, die zur Beschreibung gemeinsamer Eigenschaften von Kontexten, die jeweils bzgl. der Typen ihrer kontextbildenden Objekte übereinstimmen, dienen. Wir werden in den folgenden Abschnitten erläutern, dass Kontexte in Abhängigkeit von den Typen ihrer kontextbildenden Objekte unterschiedliche Eigenschaften besitzen. Es macht beispielsweise Sinn, in einem Kontext, der an einen Geschäftsfall gebunden ist, andere Typen von Informationen zu verwalten als in einem Kontext, der an einen Prozess gebunden ist. Der Begriff des Kontexttyps vereinfacht die Formulierung dieser Betrachtungen.

Zur begrifflichen Unterscheidung verschiedener Kontexttypen werden jeweils die zu einem Kontexttyp zugehörigen kontextbildenden Objekttypen verwendet. Beispielsweise bezeichnet ein Aktivitäts-Kontexttyp einen Kontexttyp, der durch die Bindung an den kontextbildenden Objekttyp „Aktivität“ festgelegt ist.

Definition 18: Kontextdokument

„Ein Kontextdokument ist ein Dokument, das in einem Kontext oder in mehreren Kontexten enthalten ist.“

Den Zusammenhang zwischen kontextbildenden Objekten und Kontexten auf der Instanz- und Typebene verdeutlicht Abbildung 28.

Wie aus der Definition des Kontextbegriffs hervorgeht, sind unterschiedliche Kontexte sinnvoll und notwendig, um das sekundäre Wissen in einem Unternehmen zu strukturieren und so dem Bearbeiter den Zugriff auf das für ihn jeweils relevante Wissen zu erleichtern. Eine erste Sammlung verschiedener Kontexte haben wir bereits in Abbildung 27 vorgestellt (jedes Oval, das in der Abbildung eine Menge von sekundären Dokumenten und ein Workflow-Objekt umfasst, stellt einen Kontext dar). Die Strukturierung sekundären Wissens orientiert sich dort lediglich an den kontextbildenden Objekttypen „Aktivität“, „Werkzeug“ und „Rolle“ (diese sind Teil der Prozess- bzw. Organisationsdefinition) und „Geschäftsfall“ (Workflow-Objekte, die zur Prozesslaufzeit instanziiert werden).

Die Nutzung von Kontexten durch den Bearbeiter im Rahmen der Geschäftsfalldurchführung und die Auswirkung der Nutzung auf den Prozess kann folgendermaßen skizziert werden: Ein Prozess bzw. seine Repräsentation in einem Prozessmodell stellt primäres Wissen dar (Definition 13). Bei der Instanziierung eines Prozesses, d.h. der Erzeugung eines zugehörigen Geschäftsfalls, wird das primäre Wissen durch das WFMS zur Steuerung des Geschäftsfalls genutzt, und es entsteht weiteres primäres Wissen, wie z.B. die für den speziellen Geschäftsfall erzeugten Dokumente, die Reihenfolge der durchgeführten Aktivitäten, etc. Zusätzlich kann

der Bearbeiter auf das in Kontexten repräsentierte sekundäre Wissen zugreifen, um die Durchführung des Geschäftsfalls situativ anzupassen. Diese Anpassung verläuft jedoch ausschließlich unterhalb der Granularitätsebene, die durch das Prozessmodell vorgegeben wird, d.h. aufgrund des sekundären Wissens führt der Bearbeiter z.B. eine anstehende Aktivität in besonderer Weise durch, indem er mit einem Experten kommuniziert oder bestimmte Werte in ein Formular einträgt. Die Steuerung des Prozesses durch das WFMS wird aber sowohl auf Typebene (Prozessmodell) als auch auf Instanzebene (Geschäftsfall) nicht von dieser Anpassung beeinflusst, d.h. die Aktivitäten und die Reihenfolge ihrer Durchführung, die zu bearbeitenden Dokumente, die zu beteiligten Rollen, etc. werden durch die Nutzung sekundären Wissens durch den Bearbeiter nicht verändert.

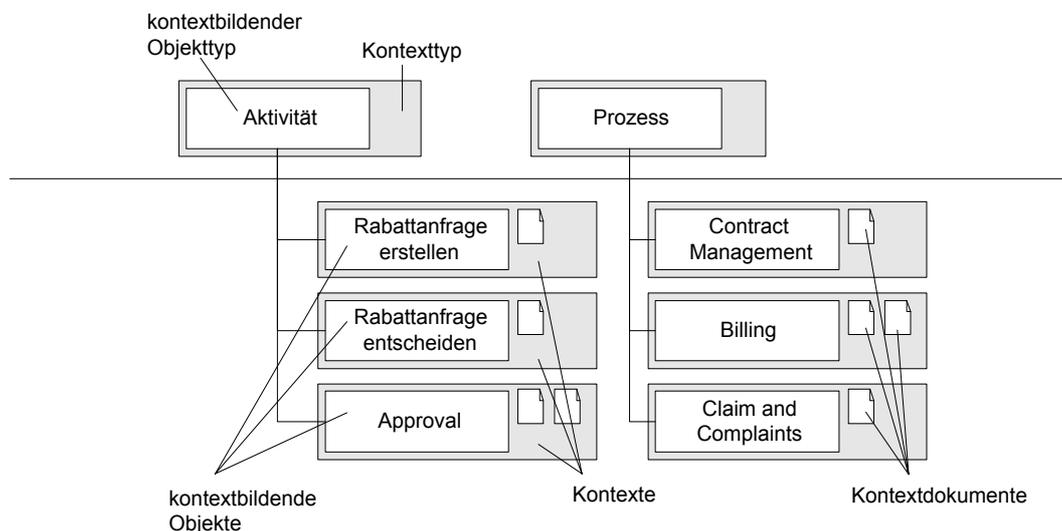


Abbildung 28: Typen und Instanzen von Kontexten und kontextbildenden Objekten

Zur schrittweisen Erstellung eines Konzepts zur Unterstützung wissensintensiver Prozesse durch Kontexte muss zunächst untersucht werden, wie ein geeigneter Strukturierungsrahmen für sekundäres Wissen anhand der kontextbildenden Objekte aussieht, d.h. welche Bearbeitungs- und Nutzungssituationen bei der Durchführung von Prozessen separate Kontexte rechtfertigen. Eine entsprechende Strukturierung sekundären Wissens aufbauend auf der Betrachtung existierender Workflow-Metamodelle wird in Abschnitt 6.2 vorgestellt.

Während WFMS in der Lage sind, primäres Wissen im Prozessmodell zu repräsentieren und zur Steuerung des Prozesses zur Laufzeit zu nutzen, fehlen darüber hinaus Möglichkeiten zur Beschreibung und Verwaltung sekundären Wissens (Definition 14). Aufbauend auf dem Konzept des Kontexts als Strukturierungsrahmen wird in Abschnitt 6.3 ein Informationsmodell für Kontextdokumente vorgestellt. Dabei werden die einzelnen Aspekte zur Beschreibung von Kontextdokumenten aus den in Kapitel 5 aufgeführten Anforderungen abgeleitet.

Die Verwaltung des gesamten prozessbezogenen Wissens muss insbesondere die vom WFMS gespeicherten Informationen zu Prozessmodellen und Laufzeitaspekten von Geschäftsfällen berücksichtigen. Abschnitt 6.4 stellt diesbezüglich eine Standardbibliothek von Systemdoku-

menten vor. Dabei handelt es sich um spezielle Kontextdokumente, die dem Bearbeiter jeweils prozessbezogene Informationen, wie z.B. einen Überblick über laufende oder abgeschlossene Geschäftsfälle, bereitstellen.

6.2 Identifikation von kontextbildenden Objekttypen

Kontexte setzen Dokumente in Beziehung zu Workflow-Objekten, um damit das sekundäre Wissen zu strukturieren und dem Bearbeiter das jeweils relevante Wissen anzubieten. Nicht jedes Workflow-Objekt ist aber sinnvollerweise kontextbildend, d.h. es dient zur Strukturierung sekundären Wissens für eine bestimmte Bearbeitungs- bzw. Nutzungssituation. So können zwar offensichtlich Aktivitäten als kontextbildende Objekte angesehen werden, indem die entsprechenden Kontexte jeweils die Dokumente enthalten, die die Bearbeitung der jeweiligen Aktivität unterstützen (vgl. Abbildung 27), bei anderen vom WFMS verwalteten Objekten, wie Zugriffsrechten oder WFMS-internen Steuerungsdaten, handelt es sich aber sinnvollerweise nicht um kontextbildende Objekte.

In diesem Abschnitt wird auf der Basis der in Abschnitt 5.3 aufgeführten Anforderungen ein Vorschlag für eine Strukturierung sekundären Wissens dargestellt, indem praxisrelevante kontextbildende Objekttypen identifiziert werden. Ziel ist es, dem Bearbeiter eine Ordnungsstruktur zur Verwaltung sekundären Wissens zur Verfügung zu stellen, die ihm bekannt ist und die ihm die Suche nach relevantem Wissen für seine aktuelle Bearbeitungssituation erleichtert. Darüber hinaus soll die Ordnungsstruktur automatisch durch das WFMS aufgebaut werden können, da die entsprechenden kontextbildenden Objekte vom WFMS verwaltet werden.

Im Folgenden werden in Abschnitt 6.2.1 zunächst existierende Metamodelle des Workflow Managements im Hinblick auf die Identifikation von kontextbildende Objekttypen untersucht. Abschnitt 6.2.2 stellt darauf aufbauend die Menge relevanter kontextbildender Objekttypen vor. Zusätzlich wird die Menge und die Struktur der aus den kontextbildenden Objekttypen abgeleiteten Kontexttypen erläutert und jeweils anhand von Verwendungsregeln für den Bearbeiter und Beispielen gezeigt, welchen Zweck ein Kontexttyp für die Bearbeitung wissensintensiver Prozesse erfüllt.

6.2.1 Untersuchung existierender Workflow-Metamodelle

Zur Identifikation kontextbildender Objekte ist es hilfreich, zunächst zu untersuchen, welche bzgl. dieser Fragestellung relevanten Workflow-Objekte die existierenden Workflow-Management-Ansätze enthalten. Wie in Definition 15 aufgezeigt wurde, ist ein Workflow-Objekt entweder im Prozess- bzw. Organisationsmodell festgelegt, oder es repräsentiert einen Geschäftsfall. Entsprechende Workflow-Objekttypen der Prozess- oder Organisationsstruktur sind in verschiedenen Workflow-Management-Ansätzen durch Metamodelle der zugehörigen Workflow-Modellierungsmethoden beschrieben. Dabei handelt es sich bei einem Metamodell einer Modellierungsmethode um einen „Gestaltungsrahmen, der die verfügbaren Arten von Modellbausteinen (Objekttypen) und die Beziehungen zwischen den Modellbausteinen zusam-

men mit ihrer Semantik festlegt sowie die Regeln für die Verwendung und Verfeinerung von Modellbausteinen und Beziehungen definiert“ [FS94]. Ein Metamodell zu einer Workflow-Modellierungsmethode wird im Folgenden kurz als Workflow-Metamodell bezeichnet.

Die existierenden Workflow-Metamodelle, die teilweise sehr umfangreich sind, müssen auf Entitäten hin untersucht werden, die zu einer Strukturierung sekundären Wissens geeignet sind. Die meisten Metamodelle folgen grundsätzlich einem sichten- bzw. aspektorientierten Ansatz, wobei der in Abschnitt 2.1.3 bereits kurz skizzierte Ansatz von Jablonski & Bußler hier als Grundlage für die Fragestellung dienen soll, welche Aspekte (bzw. Sichten) von Metamodellen in Bezug auf die hier relevante Fragestellung grundsätzlich zu untersuchen sind.

Der funktionale Aspekt sowie der organisatorische Aspekt bilden den Hauptansatzpunkt für die Identifikation kontextbildender Workflow-Objekte, da sowohl die Repräsentation von Prozessen und Aufgaben als auch die Abbildung von organisatorischen Verantwortlichkeiten zur Strukturierung sekundären Wissens geeignet erscheinen (siehe das in Abbildung 27 dargestellte Beispiel). In Bezug auf die prozessbezogenen Informationen und die verwendeten Werkzeuge sind die entsprechenden Aspekte des Metamodells nach Jablonski & Bußler (Informationsaspekt und operationaler Aspekt) nur auf sehr allgemeiner Ebene von Bedeutung. Auf einer detaillierten Ebene beschreibt der Informationsaspekt (wie in Abschnitt 2.1.3 dargestellt) Parameter, Datentypen und den Datenfluss zwischen Workflows, während der operationale Aspekt zur Definition der Schnittstelle von Werkzeugen dient. Die entsprechenden Workflow-Objekte sind für den Bearbeiter im Allgemeinen transparent und zu feingranular, um für eine Strukturierung sekundären Wissens geeignet zu sein. Daher werden diese Aspekte in den folgenden Abschnitten nicht detailliert dargestellt. Der verhaltensbezogene Aspekt schließlich, der den Kontrollfluss zwischen Workflows abbildet, ist für die zu untersuchende Fragestellung irrelevant.

Die folgenden Abschnitte haben nicht zum Ziel, eine vollständige Übersicht über alle publizierten Workflow-Metamodellansätze zu geben. Vielmehr werden einige relevante Workflow-Metamodelle präsentiert, die in Bezug auf die oben vorgestellten Aspekte detailliert sind. Zunächst wird das Metamodell der Workflow Management Coalition in Bezug auf den funktionalen Aspekt skizziert (Abschnitt 6.2.1.1). Das Workflow-Metamodell nach Galler (Abschnitt 6.2.1.2) dient zur Erläuterung des organisatorischen Aspekts. Da es sich bei diesen Ansätzen um die Beschreibung datenorientierter WFMS handelt, wird zusätzlich das Metamodell des kommerziellen WFMS CSE Workflow (Abschnitt 6.2.1.3) als Beispiel eines dokumentenorientierten WFMS vorgestellt. Wir werden in Abschnitt 6.2.2 die wesentlichen Elemente dieser Metamodelle für die Identifikation kontextbildender Objekte verwenden.

6.2.1.1 Workflow-Metamodell der Workflow Management Coalition

Abbildung 29 zeigt eine Sicht auf das Workflow-Metamodell der Workflow Management Coalition [WfMC99a, WfMC99b], wobei in der gewählten Darstellung der Schwerpunkt auf der funktionalen Sicht liegt. Ein Prozess kann nach dem Metamodell der WfMC hierarchisch aufgebaut sein, also aus Subprozessen bestehen. Zusätzlich wird in [WfMC99a] die Entität des „Activity Block“ eingeführt (dieser findet sich allerdings nicht im eigentlichen Metamodell der WfMC und ist daher in der Abbildung grau wiedergegeben). Ein „Activity Block“ bezeichnet eine Menge von Aktivitäten, die gemeinsame Eigenschaften (z.B. koordinierte Ressourcenallo-

kation) besitzen, die vom WFMS bei der Ausführung beachtet werden sollen. Eine atomare Aktivität kann mit einer Menge von Werkzeugen (über die „Workflow Application Declaration“) verknüpft sein, und die Menge der Bearbeiter wird durch die „Workflow Participant Specification“ festgelegt. In Bezug auf den organisatorischen Aspekt enthält das Workflow-Metamodell der WfMC nur sehr eingeschränkte Möglichkeiten, Bearbeiter für Aktivitäten festzulegen, wobei ein externes Organisationsmodell referenziert werden kann. Bezüglich der Möglichkeiten zur Modellierung des Informationsaspekts beschreibt das Metamodell ein datenorientiertes WFMS. Der Aufbau und die Funktion der „Workflow Relevant Data“ ist in Abschnitt 2.1.3 beschrieben.

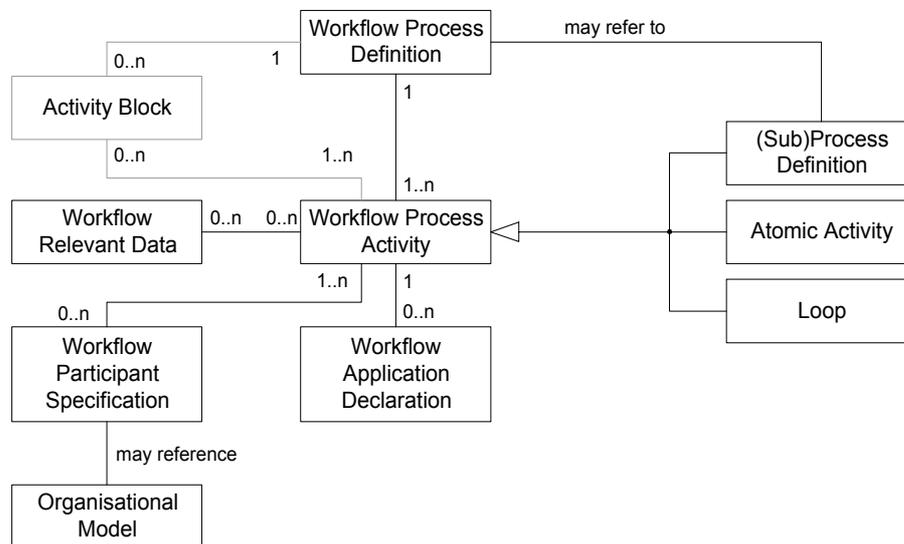


Abbildung 29: Ausschnitt aus dem Workflow-Metamodell der Workflow Management Coalition [WfMC99a, WfMC99b]

6.2.1.2 Workflow-Metamodell von Galler

Das Workflow-Metamodell von Galler bietet bezüglich des funktionalen Aspekts keine wesentlichen Neuerungen im Vergleich zum Metamodell der WfMC. Die Modellierung von prozessbezogenen Informationen folgt dem datenorientierten Ansatz und soll hier nicht erläutert werden. Darüber hinaus bietet das Metamodell ein detailliertes Organisationsmodell, das in Abbildung 30 in seinen Grundzügen dargestellt wird: Funktionen werden von organisatorischen Einheiten durchgeführt. Zu letzteren zählen Stellen, Akteure, die Stellen besetzen, und Organisationseinheiten, die hierarchisch organisiert sein können und denen Stellen zugeordnet sind. Zusätzlich kann jeder organisatorischen Einheit eine Menge von Rollen zugeteilt sein. Diese Rollen setzen bestimmte Qualifikationen voraus, über die die Akteure verfügen müssen, die eine entsprechende Rolle wahrnehmen.

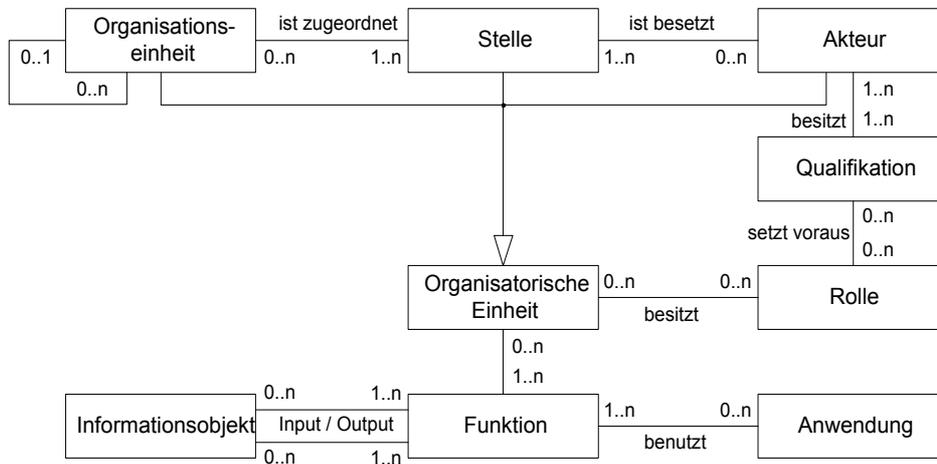


Abbildung 30: Ausschnitt aus dem Workflow-Metamodell von Galler [Gall95]

6.2.1.3 Workflow-Metamodell des Systems CSE Workflow

Abschließend soll im Folgenden das Metamodell des WFMS CSE Workflow in seinen wesentlichen Aspekten skizziert werden (Abbildung 31). Der funktionale Aspekt ist bei CSE Workflow relativ einfach gehalten (es existieren z.B. keine hierarchisch zusammengesetzten Prozesse), und das Organisationsmodell folgt im Wesentlichen den bei Galler dargestellten Aspekten. Bezüglich der Beschreibung des Informationsaspekts handelt es sich bei CSE Workflow (siehe Abschnitt 2.1.3) im Gegensatz zu den bisher dargestellten Ansätzen um ein dokumentenorientiertes WFMS. Dokumente sind jeweils über eine Dokumentvorlage mit einer Anwendung verknüpft, die die Bearbeitung des Dokuments zulässt (Bsp.: Dokument „Rabattanfrage für Fa. Siemens“, Vorlage: „Rabattanfrage“, Anwendung „MS Excel“). Dokumente werden nicht in einem Prozessmodell festgelegt und sind daher nicht direkt mit einem Prozess oder mit Aktivitäten verbunden. Lediglich die Menge der für einen Prozess verfügbaren, also bei der Bearbeitung instanziierten Dokumentvorlagen wird zum Modellierungszeitpunkt spezifiziert. Dokumente werden von den Bearbeitern zur Laufzeit in eine Umlaufmappe gelegt, die jeweils einen einzelnen Geschäftsfall repräsentiert. Um diesen Zusammenhang abbilden zu können, wurde das Workflow-Objekt „Geschäftsfall“ in die Abbildung des Metamodells aufgenommen (grau dargestellt), obwohl in der Workflow-Modellierungssprache keine entsprechenden Entitäten zur Definition von Geschäftsfällen existieren und der Geschäftsfall damit streng genommen nicht zum Workflow-Metamodell gehört.

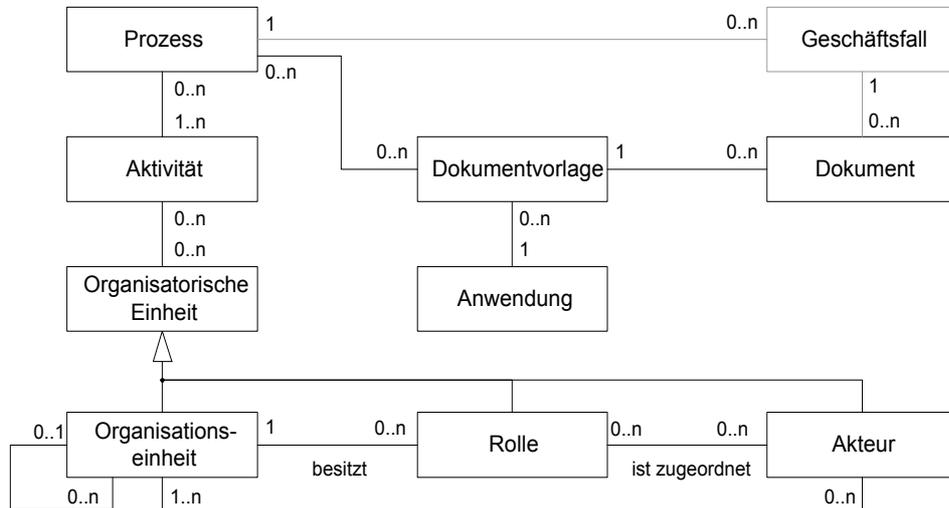


Abbildung 31: Ausschnitt aus dem Metamodell des WFMS CSE Workflow

Die skizzierten Workflow-Metamodelle enthalten die wesentlichen Entitäten, die für die Identifikation der relevanten Menge kontextbildender Objekttypen notwendig sind. Weitere Workflow-Metamodelle von Deiters [Deit93], Derungs et al. [DVÖ96], Jablonski & Bußler [JB96] und Joosten [Joos95] bringen bzgl. dieser Fragestellung keine neuen Aspekte.

6.2.2 Identifikation der kontextbildenden Objekttypen und Struktur der zugehörigen Kontexte

Aufbauend auf der Darstellung von Workflow-Metamodellen soll ein Vorschlag für die Menge der kontextbildenden Objekttypen erarbeitet werden (die Bezeichnung der Objekttypen folgt dabei den in Abschnitt 2.1.1 vorgestellten Begriffsbildungen). Dabei sind die folgenden Fragen zu beachten:

- Welche Workflow-Objekttypen tragen zu einer praxisorientierten Strukturierung sekundären Wissens bei?

Nicht alle denkbaren Workflow-Objekttypen dienen der Strukturierung sekundären Wissens, das dem Bearbeiter bei der effektiven und effizienten Durchführung von Prozessen Hilfestellung bietet. Eine erste Einschränkung bzgl. der relevanten Aspekte bzw. Sichten von Workflow-Metamodellen wurde hierzu bereits in Abschnitt 6.2.1 erläutert. Dabei ist insbesondere von Bedeutung, welche Workflow-Objekttypen (und entsprechende Workflow-Objekte) den Bearbeitern überhaupt bekannt sind. So kennen insbesondere bei einer partizipativen Einführung einer Workflow-Management-Anwendung die Bearbeiter z.B. die einzelnen Aktivitäten eines Prozesses und die zugehörigen Werkzeuge, nicht aber die beim Aufruf von Werkzeugen übergebenen Daten bzw. Datenstrukturen.

- Ist die Anzahl der Strukturierungsmerkmale für den Benutzer angemessen?

Jeder kontextbildende Objekttyp bietet für den Benutzer eine Möglichkeit, sekundäres Wissen entsprechend diesem Kriterium zu strukturieren und aufzufinden. Eine Klassifikation sekundären Wissens muss dabei so konzipiert werden, dass die Anzahl der Strukturierungsmerkmale einerseits groß genug ist, um überhaupt eine echte Strukturierung zu gewährleisten, und andererseits aber für den Bearbeiter noch überschaubar ist.

Aus der Definition des Begriffs „Workflow-Objekt“ und der Diskussion der relevanten Aspekte von Workflow-Metamodellen in Abschnitt 6.2.1 folgt, dass kontextbildende Objekttypen in den folgenden Gruppen von Workflow-Objekttypen identifiziert werden können:

- Funktionaler Aspekt von Workflow-Metamodellen:

Der grundlegende Nutzen einer Integration sekundären Wissens in die Workflow-Bearbeitung liegt in der Unterstützung des Bearbeiters bei der Durchführung von spezifischen Prozessen und Aktivitäten, wobei der Nutzungs- und Bearbeitungskontext der Wissensbestände von unterschiedlicher Granularität sein kann: Bestimmte Dokumente können z.B. bei der Bearbeitung jeder Aktivität eines Prozesses relevant sein und beziehen sich daher auf den gesamten Prozess, andere Dokumente beziehen sich dagegen auf eine konkrete Aktivität im Prozess.

Die gemäß dieser Anforderung ausgewählten kontextbildenden Objekttypen „Prozess“ und „Aktivität“ wurden bezüglich ihrer Semantik an das Workflow-Metamodell der WfMC angelehnt. Lediglich der kontextbildende Objekttyp „Aktivitätsblock“ wird hier im Unterschied zum Metamodell der WfMC als eine Menge von Aktivitäten angesehen, die nicht notwendigerweise alle demselben Prozess zugehörig sind. Durch diese Entität soll der in Abschnitt 4.1.2 diskutierten Anforderung Rechnung getragen werden, dass der Wissenstransfer zwischen Prozessen einer Prozessgruppe gefördert werden soll. Dadurch dass Wissensbestände einem Aktivitätsblock, der Aktivitäten aus mehreren Prozessen umfasst, zugeordnet werden, kann dieser geforderte Wissenstransfer unterstützt werden. Außerdem kann durch einen Aktivitätsblock ebenfalls das Konstrukt des Subprozesses abgebildet werden, in dem ein Aktivitätsblock eine im Prozessmodell als Subprozess spezifizierte Menge von Aktivitäten umfasst. Es ist daher nicht notwendig, den Objekttyp „Subprozess“ zusätzlich in die Menge der kontextbildenden Objekttypen aufzunehmen.

- Organisatorischer Aspekt von Workflow-Metamodellen:

Mit Hilfe der Strukturierung sekundären Wissens anhand von Workflow-Objekten des organisatorischen Aspekts ist es möglich, Wissen über organisatorische Einheiten zu verwalten, das zwar zur Durchführung von Prozessen relevant ist, sich aber nicht auf einen konkreten Prozess bezieht. Als kontextbildende Objekttypen wurden aus den beschriebenen Metamodellen die Objekttypen „Organisationseinheit“ (ggf. hierarchisch organisiert), „Rolle“ und „Bearbeiter“ (in obigen Metamodellen als „Akteur“ bezeichnet) ausgewählt. Beispiele für sekundäres Wissen, das sich auf kontextbildende Objekte des Organisationsaspekts bezieht, sind Vertretungsregeln, Regeln zur Arbeitsverteilung, Beschreibungen der Kompetenzen und Erfahrungen einzelner Mitarbeiter oder die Beschreibung der Funktion und der Aufgaben einer bestimmten Rolle.

- Informationsaspekt von Workflow-Metamodellen:

In Bezug auf den Informationsaspekt spielt die Unterscheidung der Beschreibungskonzepte der Modellierung nach daten- und dokumentenorientierten Systemen eine besondere Rolle. Wie in Abschnitt 2.1.3 ausgeführt und an den Workflow-Metamodellen der WfMC und von Galler ersichtlich, wird bei datenorientierten WFMS grundsätzlich einer Aktivität eine Menge von Werkzeugen zugeordnet, die auf einem spezifizierten Datenbestand arbeiten, wobei sowohl die Struktur der Daten, also die Datentypen, als auch deren Input- und Outputbeziehungen zur Aktivität in der Workflow-Modellierung festgelegt werden. Zur Strukturierung sekundären Wissens erscheint es nun nicht sinnvoll, dass Wissen einzelnen Datentypen zugeordnet werden kann, da die Struktur der im Workflow verarbeiteten Daten im Allgemeinen für den Bearbeiter transparent ist. Folglich sehen wir die entsprechenden Workflow-Objekttypen „Workflow Relevant Data“ (Metamodell der WfMC) bzw. „Informationsobjekt“ (Metamodell von Galler) nicht als kontextbildend an.

Bei dokumentenorientierten WFMS, wie z.B. CSE Workflow, werden prozessbezogene Informationen grundsätzlich in Form von Dokumenten in einer dem jeweiligen Geschäftsfall zugeordneten Umlaufmappe verwaltet und nicht in der Prozessmodellierung festgelegt. Aufbauend auf dem Workflow-Metamodell von CSE Workflow werden folgende Workflow-Objekttypen als kontextbildend angesehen: Da Dokumente lediglich bestimmten Geschäftsfällen zugeordnet werden, macht ein kontextbildender Objekttyp „Dokument“ keinen Sinn, da dadurch zwar Wissen zu einem bestimmten Dokument in einem speziellen Geschäftsfall abgebildet werden könnte, diese Strukturierung aber zu feingranular und nicht praxisgerecht wäre. Um jedoch Wissen, das sich auf die Bearbeitung von Dokumenten unabhängig von konkreten Geschäftsfällen bezieht, abbilden zu können, ist der Workflow-Objekttyp „Dokumentvorlage“ geeignet, der somit als kontextbildend identifiziert wird. Eine Dokumentvorlage ist nicht an einen konkreten Prozess gebunden, sondern steht zu „Prozess“ in einer many-to-many-Relation.

- Operativer Aspekt von Workflow-Metamodellen:

Der Workflow-Objekttyp „Werkzeug“ wird als kontextbildend festgelegt, da über diesen Objekttyp allgemeines Wissen über Werkzeuge unabhängig von ihrer konkreten Zuordnung zu Dokumenten oder Aktivitäten verwaltet werden kann. Hingegen sind die exakten Schnittstellen und Aufrufspezifika der in den Workflow eingebundenen Werkzeuge für die Strukturierung sekundären Wissens nicht von Interesse, da diese im Allgemeinen für den Bearbeiter transparent sind.

Bei datenorientierten WFMS können Werkzeuge einzelnen Aktivitäten zugeordnet werden (siehe Metamodell der WfMC), während bei dokumentenorientierten Ansätzen ein Werkzeug an eine entsprechende Dokumentvorlage gekoppelt ist (siehe Metamodell von CSE Workflow). In beiden Fällen besteht zwischen „Werkzeug“ und „Prozess“ bzw. „Aktivität“ eine many-to-many-Relation.

- Instanziierung von Geschäftsfällen:

Die bisher diskutierten Aspekte bezogen sich jeweils auf Workflow-Metamodelle und damit auf Entitäten von Prozess- bzw. Workflow-Modellierungssprachen. Dynamische Aspekte, wie die Instanziierung von Geschäftsfällen, sind nicht Teil von Workflow-Modellierungssprachen und daher auch nicht in Workflow-Metamodellen enthalten. Sie sind aber dennoch

zur Strukturierung sekundären Wissens notwendig. Daher dient der kontextbildende Objekttyp „Geschäftsfall“ zur Strukturierung von Wissen, das sich jeweils auf die Durchführung eines konkreten Geschäftsfalls bezieht.

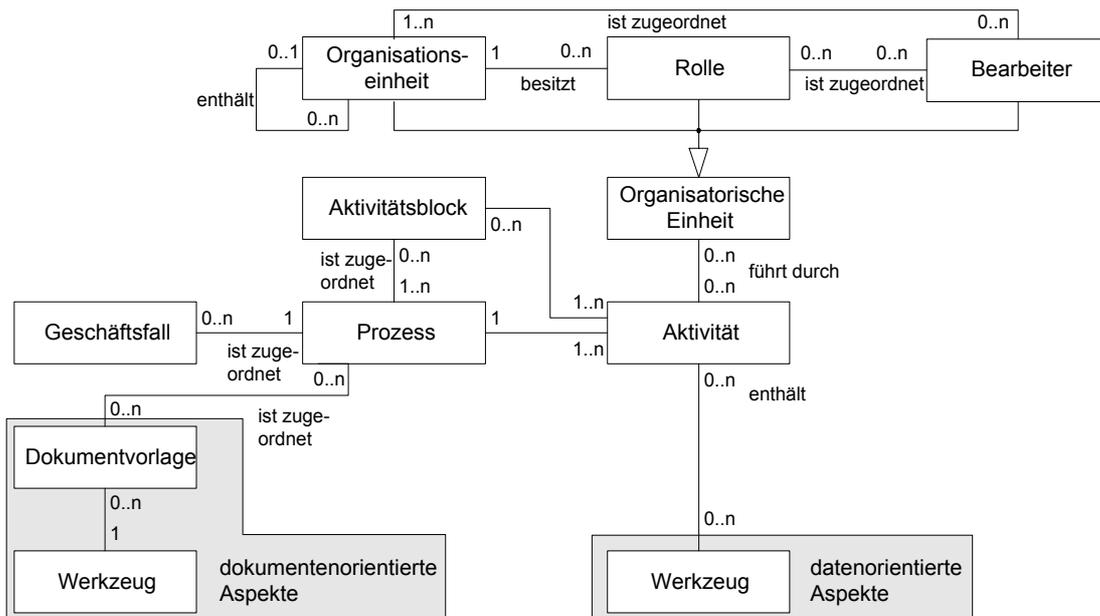


Abbildung 32: Kontextbildende Objekttypen

Abbildung 32 zeigt zusammenfassend die beschriebenen kontextbildenden Objekttypen und ihre Abhängigkeiten. Konkrete WFMS bieten jeweils nur einen Teil der hier dargestellten Objekttypen an, da sie einerseits entweder dem daten- oder dokumentenorientierten Ansatz folgen und andererseits ggf. vereinfachte Prozess- und Organisationsmodellierungsansätze beinhalten (beispielsweise unterstützt das WFMS Staffware nicht die Modellierung von Organisationseinheiten).

Durch die Identifikation der kontextbildenden Objekttypen ist ebenfalls die Menge der zugehörigen Kontexttypen festgelegt. Die jeweilige inhaltliche Beschreibung und damit auch eine Regel zur Verwendung der Kontexttypen gibt Tabelle 3 zusammen mit Beispielen für entsprechendes sekundäres Wissen an. Dabei wurden die bisher getrennt betrachteten Punkte funktionaler Aspekt, Informationsaspekt und operationaler Aspekt zum Begriff „Prozessstruktur“ zusammengefasst. Die kontextbildenden Objekttypen der Prozessstruktur legen dabei fest, für welche Bearbeitungssituation bei der Prozessbearbeitung das sekundäre Wissen relevant ist. Durch die kontextbildenden Objekttypen der Organisationsstruktur wird spezifiziert, dass sich sekundäres Wissen direkt auf bestimmte organisatorische Einheiten bezieht, und es wird die Aufgabe bzw. Funktion dieser organisatorischen Einheiten für die Prozessbearbeitung näher erläutert. Schließlich bestimmt der Aspekt der Geschäftsfallbearbeitung, ob sich das Wissen auf einzelne Geschäftsfälle bezieht oder davon unabhängig ist.

Es erscheint darüber hinaus sinnvoll, die Relevanz von Dokumenten für eine bestimmte Bearbeitungssituation nicht nur durch die Angabe eines kontextbildenden Objekttyps, sondern auch über weitere Bedingungen festzulegen. So kann bestimmtes sekundäres Wissen nur dann für die Bearbeitung eines Geschäftsfalls relevant sein, wenn dieser eine bestimmte Priorität besitzt oder für einen speziellen Kunden ausgeführt wird. Darüber hinaus ist nicht das gesamte Wissen für jeden Bearbeiter, sondern oftmals lediglich für bestimmte Adressatengruppen relevant. Wir werden diese weiteren Aspekte zur Strukturierung sekundären Wissens bei der Vorstellung eines Informationsmodells für Kontextdokumente durch die Spezifikation von Sichtbarkeitsbedingungen für Kontextdokumente (Abschnitt 6.3.5) und im Aspekt „Adressaten und Berechtigungen“ (Abschnitt 6.3.4) berücksichtigen.

Tabelle 3: Kontextbildende Objekttypen und beispielhafte Inhalte zugehöriger Kontexttypen

kontextbildender Objekttyp	Beschreibung des Inhalts der zugehörigen Kontexttypen	Beispiele für sekundäres Wissen in zugehörigen Kontexttypen
Prozessstruktur		
Prozess	Wissen mit Bezug zu einem bestimmten Prozess (relevant in allen Aktivitäten des Prozesses)	<ul style="list-style-type: none"> • Vorgaben bzgl. der Bearbeitung • Fristen • Verbesserungsvorschläge
Aktivität	Wissen mit Bezug zu einer bestimmten Aktivität (relevant bei der Bearbeitung der Aktivität)	<ul style="list-style-type: none"> • Vorgaben bzgl. der Bearbeitung (Checklisten) • Fristen • Hinweise auf Experten • Erfahrungen
Aktivitätsblock	Wissen mit Bezug zu einem bestimmten Aktivitätsblock (relevant in jeder Aktivität, die zu diesem Aktivitätsblock gehört)	<ul style="list-style-type: none"> • Prozessübergreifendes Hintergrundwissen über Wettbewerber oder Kunden
Dokumentvorlage	Wissen mit Bezug zu einer bestimmten Dokumentvorlage (relevant bei der Bearbeitung jedes Dokuments, das eine Instanz dieser Dokumentenvorlage ist)	<ul style="list-style-type: none"> • Hilfen zur Bearbeitung des Dokuments • Fehlermeldungen
Werkzeug	Wissen mit Bezug zu einem bestimmten Werkzeug (relevant bei jedem Aufruf dieses Werkzeugs)	<ul style="list-style-type: none"> • FAQ • Fehlermeldungen • Probleme und Lösungen • Hilfesysteme • Hinweise auf Experten
Instanziierung von Geschäftsfällen		
Geschäftsfall	Wissen, das bei der Bearbeitung eines bestimmten Geschäftsfalls relevant ist	<ul style="list-style-type: none"> • Gründe für Entscheidungen • Erfahrungen

Tabelle 3: Kontextbildende Objekttypen und beispielhafte Inhalte zugehöriger Kontexttypen

kontextbildender Objekttyp	Beschreibung des Inhalts der zugehörigen Kontexttypen	Beispiele für sekundäres Wissen in zugehörigen Kontexttypen
Organisationsstruktur		
Organisationseinheit	Wissen mit Bezug zu einer bestimmten Organisationseinheit	<ul style="list-style-type: none"> • Überblick über Mitarbeiter und Kompetenzen der Organisationseinheit
Rolle	Wissen mit Bezug zu einer bestimmten Rolle	<ul style="list-style-type: none"> • Vertretungsregeln • Regeln zur Arbeitsverteilung • Aufgabe und Funktion der Rolle
Bearbeiter	Wissen mit Bezug zu einem bestimmten Bearbeiter	<ul style="list-style-type: none"> • Kompetenzen und Erfahrungen eines Mitarbeiters

Mit Hilfe der identifizierten kontextbildenden Objekttypen kann das Konzept der Kontexte weiter ausgearbeitet werden: Bisher wurden lediglich Kontexttypen betrachtet, die jeweils durch die Bindung an einen einzigen kontextbildenden Objekttyp definiert wurden. Da es sich bei den beiden Aspekten „Prozessstruktur“ und „Instanziierung von Geschäftsfällen“ aber bzgl. der Strukturierung sekundären Wissens um unabhängig zu betrachtende Aspekte handelt, können durch Kombination dieser beiden Aspekte weitere praxisrelevante Kontexttypen gebildet werden. Somit kann zusätzlich unterschieden werden, ob sich sekundäres Wissen auf einen konkreten Geschäftsfall bezieht oder davon unabhängig ist, also in allen Geschäftsfällen eines Prozesses relevant sein kann (Tabelle 4).

Dabei sind die Kontexttypen zu den kontextbildenden Objekttypen „Aktivitätsblock“, „Dokumentvorlage“ und „Werkzeug“ aus folgenden Gründen nur unabhängig von konkreten Geschäftsfällen zulässig: Die Menge an sekundärem Wissen, die sich auf einen bestimmten Geschäftsfall bezieht, wird als nicht so umfangreich eingeschätzt, dass außer der Zuordnung zu einzelnen Aktivitäten weitere Strukturierungsmerkmale notwendig sind. Darüber hinaus sollen die kontextbildenden Objekttypen „Dokumentvorlage“ und „Werkzeuge“ es ermöglichen, allgemeines Wissen zu Dokumentvorlagen bzw. Werkzeugen unabhängig von konkreten Prozessausführungen zu verwalten (siehe Abbildung 32 und die Beschreibung der kontextbildenden Objekttypen).

Tabelle 4: Übersicht über Kontexttypen und Regeln zur Anwendung

Kontextbildende Objekttypen	unabhängig von konkreten Geschäftsfällen	Geschäftsfall
Prozess	Wissen mit Bezug zu einem bestimmten Prozess	Wissen mit Bezug zu einem bestimmten Prozess, eingeschränkt relevant für einen konkreten Geschäftsfall
Aktivität	Wissen mit Bezug zu einer bestimmten Aktivität	Wissen mit Bezug zu einer bestimmten Aktivität, eingeschränkt relevant für einen konkreten Geschäftsfall
Aktivitätsblock	Wissen mit Bezug zu einem bestimmten Aktivitätsblock	Diese Kontexttypen existieren nicht (siehe Erläuterung im Text).
Dokumentvorlage	Wissen mit Bezug zu einer bestimmten Dokumentvorlage	
Werkzeug	Wissen mit Bezug zu einem bestimmten Werkzeug	
Organisationseinheit	Wissen mit Bezug zu einem bestimmten Bearbeiter	
Rolle	Wissen mit Bezug zu einer bestimmten Rolle	
Bearbeiter	Wissen mit Bezug zu einer bestimmten Organisationseinheit	

Bisher wurden lediglich Kontexttypen betrachtet. Zur Verwaltung sekundären Wissens zur Laufzeit müssen den Bearbeitern aber konkrete Kontexte zur Verfügung stehen, die durch die existierenden kontextbildenden Objekte gebildet werden. Zu jedem Kontexttyp wird also bei der Bearbeitung von Prozessen eine Menge von Kontexten gebildet. Abbildung 33 zeigt hierzu exemplarisch mögliche Kontexte zu einem Kontexttyp, der durch die kontextbildenden Objekttypen „Aktivität“ und „Geschäftsfall“ gebildet wird.

Die Menge der Kontexte, die unabhängig von konkreten Geschäftsfällen sind, ist statisch zur Laufzeit, d.h. alle Kontexte können aus den im WFMS hinterlegten Prozessmodellen und Organisationsmodellen abgeleitet werden (es entsteht selbstverständlich eine veränderte Menge von Kontexten, falls Prozessmodelle und das Organisationsmodell im Rahmen des kontinuierlichen Verbesserungsprozesses verändert werden). Dagegen verändert sich die Menge der Kontexte, die sich auf konkrete Geschäftsfälle beziehen, dynamisch zur Laufzeit. Werden neue Geschäftsfälle instanziiert, bedingt dies die Entstehung neuer Kontexte (im in Abbildung 33 präsentierten Beispiel würde die Tabelle um eine Spalte erweitert). Werden Geschäftsfälle beendet, sollen die entsprechenden Kontexte allerdings nicht gelöscht werden, um auf die entsprechenden Wissensbestände auch zu einem späteren Zeitpunkt noch zugreifen zu können. Eine Versionierung von Kontexten ist denkbar, wird aber im Folgenden nicht betrachtet.

Bearbeiter sollen Kontexte bei der Prozessbearbeitung integriert in die Durchführung von Aktivitäten zugreifen und verändern können. Dabei ist zu betrachten, welche Operationen von welchen Benutzern auf Kontexten ausführbar sind. Die Erzeugung von Kontexten, wie oben beschrieben, wird durch das WFMS vorgenommen, ohne dass die Bearbeiter in diesen Prozess einbezogen werden müssen. Das Entfernen von Kontexten soll ebenfalls nicht von Bearbeitern durchgeführt werden. Diese Operation liegt in der Verantwortung des Prozessverantwortlichen bzw. des Modellierers, falls z.B. Prozessreorganisationen durchgeführt werden. Dagegen ist das Einfügen von Dokumenten in Kontexte eine Operation, die jeder Bearbeiter auf allen verfügbaren Kontexten ausüben darf. Bei der Diskussion von Adressatengruppen zu Kontextdokumenten in Abschnitt 6.3.4 werden wir aber zusätzlich die Möglichkeit vorsehen, dass einzelne Dokumente nur bestimmten Bearbeitern, Rollen oder Organisationseinheiten sichtbar sind. Mit diesen Maßnahmen wird einerseits ein möglichst großer Wissenstransfer bewirkt, indem das Erzeugen neuer Dokumente in allen Kontexten erlaubt ist, andererseits ermöglicht die eingeschränkte Sichtbarkeit von Dokumenten persönliche oder nur eingeschränkten Benutzergruppen sichtbare Informationen.

	Geschäftsfall: Fa. Siemens	Geschäftsfall: Fraunhofer-Gesellschaft	Geschäftsfall: comdirect bank
Aktivität: Rabattierung mit Kunden verhandeln		Einschätzung des Wissens über Wettbewerber	Erfolg der Verhandlungs- strategie Einschätzung des Verhandlungspart- ners Dr. Müller
Aktivität: Rabattanfrage erstellen	Gründe für abweichende Rabattierung	Gründe für abweichende Rabattierung	
Aktivität: Rabattanfrage entscheiden	Begründung der Genehmigung	Begründung der Ablehnung Verweis auf ähnliche Geschäftsfälle	

 Kontext
  Kontextdokument

Abbildung 33: Exemplarische Kontexte zu einem durch Aktivität und Geschäftsfall festgelegten Kontexttyp

Die Wissensbewahrung in Kontexten soll dadurch unterstützt werden, dass zu jedem Kontext ein Kontextverantwortlicher angegeben werden kann. Der entsprechende Bearbeiter übernimmt damit die Aufgabe, den Inhalt des Kontexts regelmäßig in Bezug auf die Aktualität zu prüfen und bei Bedarf kooperativ mit den Verantwortlichen für die einzelnen Kontextdokumente (Abschnitt 6.3.3) Änderungen anzustoßen. Eine dezidierte Verantwortlichkeit für die Wissensbewahrung in Kontexten ist nicht für alle Kontexte notwendig. Erst wenn eine

bestimmte Menge von Informationen durch einen Kontext verwaltet wird und die Informationen häufigen Änderungen unterworfen sind, ist die Einführung dieser Rolle für einen Kontext sinnvoll. Diese organisatorischen Fragestellungen werden in Kapitel 7 untersucht.

Prinzipiell besteht bei einer komplexen Strukturierung von Informationen stets die Gefahr, dass die Benutzer nicht die jeweils relevanten Informationen effizient auffinden und damit die Motivation schwindet, das System weiter zu benutzen und insbesondere aktiv weiteren Inhalt bereitzustellen. Wir halten die vorgeschlagene Strukturierung von sekundärem Wissen aus den folgenden Gründen für erfolgversprechend:

- Die kontextbildenden Objekte sind den Bearbeitern bekannt.

Die kontextbildenden Objekttypen wurden so ausgewählt, dass die entsprechenden kontextbildenden Objekte (und damit die Kontexte) den Bearbeitern bekannt sind, da sich die Bearbeiter in der für sie relevanten Prozess- und Organisationsstruktur im gewählten Detaillierungsgrad auskennen. Daher sind auch die Regeln zur Anwendung spezifischer Kontexte (Tabelle 4) leicht erlernbar. Die hier vorgeschlagene Strukturierung unterscheidet sich somit von den meisten Ordnungsstrukturen, die auf der Basis inhaltlicher Klassifikationen von Dokumenten arbeiten. Die Praxis zeigt, dass sich bei diesen Ansätzen, wie z.B. der Klassifikation nach Schlagworten in Dokumenten-Management-Systemen, oftmals Schwierigkeiten beim Aufbau einer Ordnungsstruktur ergeben, die einerseits hinreichend komplex und trennscharf ist und andererseits den Benutzern trotzdem vermittelbar ist.

- Die Anzahl der in einer konkreten Bearbeitungssituation direkt relevanten Kontexte ist begrenzt.

Für eine konkrete Bearbeitungssituation ist nur eine relativ begrenzte Menge von Kontexten für den Bearbeiter direkt relevant. Es sind dies die Kontexte zum aktuellen Prozess, zur aktuellen Aktivität und ggf. Aktivitätsblock und zum aktuell bearbeiteten Geschäftsfall. Der Inhalt dieser Kontexte muss direkt für den Bearbeiter erreichbar sein. Dagegen sind die Inhalte von Kontexten zu Werkzeugen, Dokumentvorlagen und zur Organisationsstruktur nicht notwendigerweise spezifisch für die aktuelle Bearbeitung, und sie enthalten daher mit einer geringeren Wahrscheinlichkeit relevantes Wissen für die aktuellen Bearbeitungssituation.

- Eine geeignete Präsentation von Kontexten erleichtert die Navigation und das Auffinden von relevanten Inhalten.

Die Präsentation von sekundärem Wissen muss unter Berücksichtigung der in der aktuellen Bearbeitungssituation relevanten Kontexte nach den obigen Gesichtspunkten erfolgen. Ein Konzept zur Präsentation von Kontexten wird in Abschnitt 8.2.2 vorgestellt.

- Das Informationsmodell für Kontextdokumente unterstützt die Bereitstellung und Verwaltung relevanten Wissens.

Das im folgenden Abschnitt vorgestellte Informationsmodell für Kontextdokumente enthält Aspekte, die eine weitere Unterstützung bei der Bereitstellung und dem Retrieval von relevantem Wissen geben. Dazu zählt die aktive Bereitstellung von Kontextdokumenten durch das System, die Angabe von Regeln, die die Sichtbarkeit von Kontextdokumenten abhängig vom Status von Geschäftsfällen festlegen, oder die Angabe von Informationen, die den Lebenszyklus von Kontextdokumenten betreffen.

6.3 Ein Informationsmodell zur Beschreibung von Kontextdokumenten

Zur weiteren Beschreibung von Kontexten ist eine Spezifikation der Kontextdokumente, die das sekundäre Wissen in einem prozessbezogenen OMIS repräsentieren, notwendig. Im Folgenden stellen wir ein Informationsmodell für Kontextdokumente vor, dessen einzelne Aspekte direkt aus den Anforderungen an eine Unterstützung wissensintensiver Prozesse in Kapitel 5 abgeleitet werden. Diese Aspekte dienen zur Strukturierung der Attribute von Kontextdokumenten, die in den folgenden Abschnitten detailliert vorgestellt werden.

In Abschnitt 5.1 wird eine Integration prozessbezogenen Wissens aus unterschiedlichen Quellen und insbesondere dem WFMS gefordert. Dieser Anforderung wird durch den Aspekt „Identifikation“ (Abschnitt 6.3.1) Rechnung getragen. Bach et al. unterscheiden in Bezug auf die Attribute von Wissensobjekten einerseits das „Profil“, das sich auf den Inhalt bezieht, und andererseits Informationen zum Lebenszyklus des Wissensobjekts [BVÖ99]. Wir folgen diesem Ansatz grundsätzlich und beschreiben im Aspekt „Inhalt“ (Abschnitt 6.3.2) Attribute, die den Inhalt eines Dokuments repräsentieren, und im Aspekt „Lebenszyklus“ (Abschnitt 6.3.3) gehen wir auf den kooperativen Aufbau und die Bewahrung sekundären Wissens ein. Darüber hinaus sollte die Menge der Adressaten angegeben werden können, für die das Wissen relevant ist, und der Zugriff auf sekundäres Wissen eingeschränkt werden können. Entsprechende Möglichkeiten sieht der Aspekt „Adressaten und Berechtigungen“ (Abschnitt 6.3.4) vor. Bei der Diskussion unterschiedlicher Funktionen und Charakteristik prozessbezogenen Wissens (Abschnitt 5.2) wurde diskutiert, dass Wissen zu bestimmten Zeitpunkten aktiv dem Bearbeiter angeboten werden soll bzw. nur unter bestimmten Gegebenheiten, z.B. abhängig vom Status des aktiv bearbeiteten Geschäftsfalls, relevant ist. Diese beiden Fragestellungen werden durch die Aspekte „Sichtbarkeit“ (Abschnitt 6.3.5) und „Bereitstellung“ (Abschnitt 6.3.6) adressiert.

Es handelt sich bei dem im Folgenden dargestellten Informationsmodell um einen Vorschlag, der um weitere Attribute ergänzt werden kann. Insbesondere kommerzielle Systeme bieten z.B. für den Aspekt „Inhalt“ oder „Berechtigungen“ eine Vielzahl von Attributen zur Beschreibung von Dokumenten an. Eine vollständige Beschreibung aller im ein oder anderen Anwendungsfall sinnvollen Attribute liegt jedoch nicht im Fokus dieser Arbeit. Vielmehr soll durch die einzelnen Aspekte und die ihnen zugeordneten Attribute ein grundsätzliches Informationsmodell vorgestellt werden, das die Beschreibung von Kontextdokumenten zur Unterstützung wissensintensiver Prozesse erlaubt.

6.3.1 Identifikation

Wie in Abbildung 25 dargestellt, können sich Kontextdokumente in unterschiedlichen Wissensquellen befinden. Dabei können sie selbstverständlich nur Wissen repräsentieren, das bereits als explizites Wissen vorliegt oder expliziert werden kann. Relevantes implizites Wissen muss daher zunächst expliziert, d.h. in Form von Dokumenten repräsentiert werden, um einem Kontext zugeordnet werden zu können. Darüber hinaus ist die Dokumentation von Metawissen, also Wissen über Wissen, beispielsweise in Form von Hinweisen auf Experten durch Kontextdokumente möglich.

Der Dokumentbegriff als „eine Menge von Informationen, die im Rahmen eines Prozesses benötigt oder bearbeitet werden“ (Definition 3) wurde bewusst weit gefasst, um die gesamten prozessbezogenen Informationen einheitlich beschreiben zu können. Da sich diese prozessbezogenen Informationen aber in Informationssystemen ganz unterschiedlicher Charakteristik und mit verschiedenen Schnittstellen, wie z.B. Groupware, ERP-Systeme oder branchenspezifische Individualsoftware, sowie im Dateisystem befinden können, ist ein einheitlicher Zugriff auf prozessbezogene Informationen nicht möglich.

Als ein Standard für die Identifikation von Ressourcen spielt URI/URL (Uniform Resource Identifier/Uniform Resource Locator) [BFIM98] eine immer größere Rolle für den Zugriff auf Informationen in Softwaresystemen. Die Identifikation von Dateien und Intranet- bzw. Internet-Ressourcen ist damit über einen einheitlichen Mechanismus möglich. Darüber hinaus ist festzustellen, dass mehr und mehr Softwaresysteme „internet-fähig“ werden und dabei insbesondere die Adressierung von Informationen per URL erlauben. Zu nennen sind hier z.B. die beiden Marktführer im Bereich Groupware Lotus Domino und Microsoft Exchange 2000 oder Dokumenten-Management-Systeme.

Bei der Identifikation von Kontextdokumenten müssen abhängig von der Charakteristik der Schnittstelle, falls sich das Kontextdokument unter Verwaltung eines Informationssystems befindet, zwei Fälle unterschieden werden:

- Das Kontextdokument kann über eine URL adressiert werden:
In diesem Fall wird die entsprechende URL angegeben.
- Das Kontextdokument kann nicht per URL adressiert werden:
In diesem Fall kann das Kontextdokument nicht direkt angegeben werden, und es wird stattdessen ein Kommandozeilenaufruf spezifiziert. Dieser kann zu einem ggf. parametrisierten Aufruf des entsprechenden Informationssystems führen (somit kann zumindest die Wissensquelle angegeben werden, auch wenn das konkrete Kontextdokument nicht identifiziert werden kann). Alternativ kann der Kommandozeilenaufruf zum Start eines externen Programms führen, das speziell auf die Schnittstelle des Informationssystems ausgelegt ist und die relevanten Informationen zugreifen kann. Darüber hinaus kann mit Hilfe dieser Adressierung auf einfache Art eine externe Komponente eingebunden werden, die Abfragen auf einem Dokumentenbestand ausführt und die Ergebnisse dem Bearbeiter anzeigt.

Tabelle 5 fasst die oben beschriebenen Sachverhalte noch einmal zusammen und stellt das zugehörige Attribut zur Identifikation von Kontextdokumenten anhand der Eigenschaften „Attributname“, „Bezug des Attributs“, „Datentyp“ und „Wertemenge/Plausibilität“ dar. Der Bezug des Attributs gibt dabei an, ob es sich um ein dokumentbezogenes oder ein kontextbezogenes Attribut handelt. Dokumentbezogene Attribute beschreiben Eigenschaften, die sich auf das Dokument unabhängig von einer Bindung in Kontexten beziehen, während kontextbezogene Attribute die Bindung eines Dokuments in einem konkreten Kontext beschreiben und damit insbesondere den Bezug eines Dokuments zu einer bestimmten Bearbeitungssituation spezifizieren.

Tabelle 5: Attribute des Aspekts „Identifikation“

Attributname	Bezogen auf	Datentyp	Wertemenge/Plausibilität
Identifikation	Dokument	Text	Adressierung per URL: Wert muss gültige URL darstellen. Adressierung per Programmaufruf: Wert muss gültiger Kommandozeilenaufruf sein.

6.3.2 Inhalt

In der Literatur existieren eine Reihe von Standards und Referenzmodelle für Metainformationen sowie insbesondere für die inhaltliche Beschreibung von Ressourcen (eine Übersicht gibt [Pfen00]). Die folgenden Attribute wurden einerseits aus der Betrachtung theoretischer Ansätze zur Beschreibung von Wissensobjekten [BVÖ99] und Metadatenstandards (hier erscheint insbesondere das Dublin Core Metadata Element Set [DCME99] geeignet) gewonnen. Andererseits lieferte die Untersuchung existierender Wissensmanagementsysteme weitere relevante Aspekte (diesbezüglich wurden Open Text Livelink [Open99], Hyperwave Information Server [Hype00] und WIS [Luca01], ein von der Württembergischen Versicherung entwickeltes und vom Fraunhofer ISST in Projekten eingesetztes Intranet-System, betrachtet). Die als Ergebnis dieser Untersuchung identifizierten Attribute werden im Weiteren beschrieben.

- Titel
Die Bezeichnung des Dokuments
- Beschreibung
Eine inhaltliche Beschreibung des Dokuments, wie z.B. ein Abstrakt, das Inhaltsverzeichnis oder eine Freitextbeschreibung des Dokumenteninhalts
- Stichworte
Stichworte dienen zur weiteren inhaltlichen Beschreibung des Dokuments gemäß vorgegebenen Kategorien.
- Format
Das Format gibt den Medientyp des Kontextdokuments an, um eine Verbindung zu einem Softwaresystem herzustellen, das die Anzeige oder die Modifikation des Dokuments ermöglicht. Der Wertebereich dieses Attributs ist durch die Liste der MIME-Types vorgegeben.

Die bisher aufgeführten Attribute dienen zur inhaltlichen Beschreibung des Dokuments und sind also dokumentbezogen. Dagegen geben die folgenden kontextbezogenen Attribute eine inhaltliche Charakterisierung eines Dokuments für einen konkreten Kontext an.

- Typ

Während Stichworte zur inhaltlichen Beschreibung dienen, wird über den Typ die Funktion, die das Dokument innerhalb der Prozessbearbeitung in einem speziellen Kontext erfüllt, spezifiziert. Aufbauend auf Abbildung 26 beschreiben wir eine Menge von Standardtypen, die dem Bearbeiter zur Auswahl angeboten werden (Tabelle 6). Die Liste der Typen soll prinzipiell durch den Administrator des Systems erweiterbar sein.

Tabelle 6: Standardtypen von Kontextdokumenten

Typ	Beschreibung
Systemdokument	Prozessmodell- oder Laufzeitinformationen, die aus dem WFMS stammen und in Form von Dokumenten aufbereitet werden (Abschnitt 6.4)
Ziel	Beschreibung von Zielen bzgl. der Bearbeitung von Prozessen, Aktivitäten oder einzelnen Geschäftsfällen
Hilfe	Informationen über z.B. Frequently Asked Questions oder Hilfesysteme zu Werkzeugen
Kommunikationsprotokoll	Protokoll der Kommunikation mit z.B. Experten oder externen Partnern
Person	Hinweis auf z.B. einen Experten und Kontaktinformationen (E-Mail, WWW, Telefon, Adresse, etc.)
Vorgabe	Vorgaben bzgl. der Bearbeitung, wie z.B. Checklisten
Bearbeitungsfristen	Angaben von Bearbeitungsfristen gehören eigentlich zu Vorgaben. Aufgrund der besonderen Relevanz von zeitlichen Vorgaben, ist aber ein separater Typ sinnvoll.
Erfahrung	Explizierung von Erfahrungen, die im Bezug zur Prozessbearbeitung stehen
Gründe für Entscheidung	Gründe für Entscheidungen, die während der Bearbeitung getroffen wurden und nachvollziehbar sein sollen.
Fehlermeldung	Fehler bzgl. des WFMS oder einzelner Dokumente, Werkzeuge, etc.
Verbesserungsvorschlag	Ideen zur Verbesserung des Prozesses auf organisatorischer oder technischer Ebene
Hintergrundinformation	Allgemeine Hintergrundinformation bzgl. der Bearbeitung, die nicht einer der obigen Klassen zugeordnet werden kann

- Funktion

Während das Attribut „Beschreibung“ den Inhalt des Dokuments zusammenfasst, gibt das Attribut „Funktion“ eine textuelle Beschreibung der Aufgabe, die das Dokument in einem speziellen Kontext erfüllt, und charakterisiert dadurch den Typ des Dokuments näher.

- Relationen

Relationen bilden Referenzen auf Dokumente ab, die mit dem betrachteten Dokument in Beziehung stehen. Dokumente, die in Relation stehen, müssen sich nicht notwendigerweise im selben Kontext befinden.

- Relevanz

Um eine optimale Wissensnutzung gewährleisten zu können, ist es wichtig, dass Bearbeiter auf einfache Weise die inhaltliche Relevanz von Dokumenten im zugehörigen Kontext auf einer ordinalen Skala bewerten können (0 = „nicht relevant in diesem Kontext“ bis 6 = „notwendig zur Bearbeitung“). Diese Bewertung soll also nicht nur vom Ersteller (siehe Abschnitt 6.3.3) des Dokuments, sondern von allen Bearbeitern fortlaufend über die Nutzung abgegeben werden können und durch die Möglichkeit zur Abgabe von Kommentaren unterstützt werden können.

Zusätzliche Kommentare in textueller Form geben konkretere Hinweise für den weiteren Umgang mit dem Dokument. Sowohl die numerische Wertung der Relevanz als auch Kommentare können damit die Grundlage für spätere Aktionen bilden, die den Lebenszyklus des Dokuments betreffen, wie z.B. das Löschen des Dokuments aus dem Kontext, die Übertragung in einen anderen, besser geeigneten Kontext oder die Übertragung des Dokuments in das Prozessmodell, falls eine besonders hohe Relevanz des Dokuments durch die Bearbeiter festgestellt wird (Überlegungen zur Evaluation von Kontextdokumenten finden sich in Abschnitt 7.1.3). Da sich aber sowohl Relevanz als auch Kommentare primär auf den Dokumenteninhalt beziehen, werden die Attribute an dieser Stelle zum Aspekt „Inhalt“ ausgeführt.

Tabelle 7 stellt die Attribute des Aspekts „Inhalt“ zusammenfassend dar. Da in diesem Kapitel die notwendigen Datenstrukturen für Kontextdokumente nicht auf der Ebene des Softwaredesigns entwickelt werden sollen, verwenden wir zur Beschreibung von Beziehungen zwischen Objekten den Datentyp „Referenz“. Dadurch abstrahieren wir von den unterschiedlichen Möglichkeiten, Objekte zu speichern und über Schlüssel zu identifizieren.

Tabelle 7: Attribute des Aspekts „Inhalt“

Attributname	Bezogen auf	Datentyp	Wertemenge/Plausibilität
Titel	Dokument	Text	keine Einschränkung
Beschreibung	Dokument	Text	keine Einschränkung
Stichworte	Dokument	Menge von Referenz	Jeder Wert muss ein vorhandenes Stichwort referenzieren.
Format	Dokument	Aufzählungstyp	MIME-Types
Typ	Kontext	Aufzählungstyp	keine Einschränkung
Funktion	Kontext	Text	keine Einschränkung

Tabelle 7: Attribute des Aspekts „Inhalt“

Attributname	Bezogen auf	Datentyp	Wertemenge/Plausibilität
Relationen	Kontext	Menge von Referenz	Jeder Wert muss auf ein Kontextdokument verweisen.
Relevanz	Kontext	Menge von Aufzählungstyp und Text	[0..6] keine Einschränkung

6.3.3 Lebenszyklus

Um den kooperativen Aufbau und die Bewahrung von Wissen zu unterstützen, sind Attribute erforderlich, die den Lebenszyklus von Kontextdokumenten beschreiben. Entsprechende Attribute umfassen sowohl die Historie eines Kontextdokuments als auch eine Beschreibung von Verantwortlichkeiten für das Dokument und zukünftig auszuführende Operationen.

Prinzipiell kann der Lebenszyklus von Dokumenten durch die Spezifikation von entsprechenden Workflows geregelt werden. Dokumenten-Management-Systeme oder Content-Management-Systeme enthalten oftmals Funktionalitäten, die den Lebenszyklus von Dokumenten (insbesondere der Prozess von der initialen Erstellung bis zur Freigabe des Dokuments ggf. durch unterschiedliche Rollen ist hier von besonderer Relevanz) per Workflow Management steuern. Diese Möglichkeiten zur exakten Steuerung z.B. der Freigabe sind für das Management von Kontextdokumenten im Allgemeinen nicht notwendig. Während es sich z.B. bei der durch ein Content-Management-System unterstützten kontinuierlichen Aufbereitung des Internet-Auftritts einer Organisation um unternehmenskritische Informationen handelt, deren Freigabe exakt sichergestellt sein muss, ist beim kooperativen Aufbau sekundären Wissens durch die Mitarbeiter ein weniger restriktives Vorgehen vorteilhaft. Die Tatsache, dass jeder Mitarbeiter Kontexte verändern kann, indem er z.B. neue Kontextdokumente einstellt, und dass diese Änderungen ohne langwierige Freigabeverfahren direkt sichtbar werden, wirkt sich dabei motivationsfördernd aus. Restriktive Regelungsmechanismen sind lediglich für das Management des primären Wissens unabdingbar. Dieses ist im Prozessmodell festgelegt, und die Änderung des Prozessmodells wird durch die Regelungen des kontinuierlichen Verbesserungsprozesses sinnvollerweise auf eine ausgewählte Personengruppe beschränkt. Ein weniger restriktives Vorgehen, das einerseits dem Mitarbeiter größere Freiheiten lässt und andererseits Lebenszyklusinformationen von Kontextdokumenten unterstützt, soll durch die folgenden Attribute gewährleistet werden:

- Exaktheit

Um einen kooperativen Aufbau von Wissen zu ermöglichen, muss der Lebenszyklus von sekundärem Wissen ausgehend von der Identifikation eines Wissensbedarfs bis zu einer exakten Spezifikation eines Kontextdokuments zur Deckung eines Wissensbedarfs unterstützt werden. Dabei tritt häufig der Fall ein, dass nach der Äußerung eines Wissensbedarfs zunächst eine vage Beschreibung von entsprechendem relevanten Wissen erfolgt, da dem Mitarbeiter z.B. zwar die Wissensquelle bekannt ist, nicht aber die exakten Dokumente, und

erst später von einem anderen Bearbeiter die Dokumente konkret angegeben werden. Daher geben wir vier Stufen an, mit denen die Exaktheit der Beschreibung klassifiziert werden kann:

Stufe 1 - Wissensbedarf:

Es wird ein Wissensbedarf dargestellt, ohne dass dem Bearbeiter aber Hinweise auf entsprechende Dokumente zur Erfüllung des Wissensbedarfs vorliegen (Bsp.: „Zur Abschätzung des Aufwands wären entsprechende Informationen zu Referenzprojekten hilfreich“).

Stufe 2 - Allgemeine Beschreibung:

Es werden Hinweise zur möglichen Identifikation relevanten Wissens gegeben. Die Hinweise sind aber vage, d.h. der Wissensbedarf ist durch diese Information noch nicht erfüllt, und es kann weder eine konkrete Wissensquelle noch ein konkretes Dokument angegeben werden (Bsp.: „Das Projekt MOVE könnte ein geeignetes Referenzprojekt sein. Es gibt dazu wohl einige Excel-Tabellen“).

Stufe 3 - Beschreibung der Wissensquelle:

Es kann eine relevante Wissensquelle angegeben werden, ohne dass aber spezifische Dokumente bekannt sind. Beim Zugriff auf die Wissensquelle ist also noch weiterer Suchaufwand erforderlich (Bsp.: „Im DMS gibt es ein Projekt MOVE und ein Schlagwort Projektkalkulation. Die relevanten Dokumente müssten darüber zu finden sein“).

Stufe 4 - Exakte Beschreibung:

Es wird ein Dokument spezifiziert, das den Wissensbedarf erfüllt.

- Ersteller
Name der Person, die das Kontextdokument in den Kontext eingebracht hat
- Verantwortlicher (Dokumentverantwortlicher)
Name der Person, die für das Kontextdokument verantwortlich ist, und damit regelmäßige Aktualisierungen des Inhalts oder das Löschen des Dokuments aus dem Kontext durchführen muss
- Autor
Name der Personen, die als Autoren des Kontextdokuments fungieren. Für den Fall, dass ein bestehendes Dokument in einen Kontext eingebracht wird, ist der Ersteller des Kontextdokuments nicht notwendigerweise einer der Autoren.
- Gültigkeitszeitraum
Angabe des Zeitraums, für den das Kontextdokument gültig ist und damit für die Bearbeiter sichtbar und zugreifbar sein soll. Es wird das Anfangsdatum und das Enddatum des Gültigkeitszeitraums angegeben. Ist das Enddatum überschritten, ist das Dokument nur noch für den Kontextverantwortlichen und den Dokumentverantwortlichen sichtbar.
- Revisionsrhythmus
Dieses Attribut gibt den Zeitraum an, mit dem das Dokument normalerweise aktualisiert wird [BVÖ99]. Mögliche Ausprägungen sind „täglich“, „wöchentlich“, „monatlich“, „jährlich“, „niemals“ und „unbekannter Revisionsrhythmus“. Der Attributwert gibt damit eine Hilfestellung für die Person, die die Rolle „Verantwortlicher“ übernimmt, um in regelmäßi-

gen Abständen die Aktualität des Dokuments zu prüfen und ggf. Änderungen vorzunehmen oder das Dokument aus dem Kontext zu löschen. Wir werden in Abschnitt 7.1.2 auf diesen Aspekt näher eingehen.

- Historie

Das Attribut Historie wird vom System benutzt, um eine Historie von Aktionen auf dem Kontextdokument zu speichern. Zu jeder Operation, die ein Bearbeiter auf dem Kontextdokument ausführt, wird der Operationstyp (Erzeugen, Öffnen, Speichern, etc.), der Name des Bearbeiters und der Zeitpunkt vermerkt. Mit Hilfe dieser historischen Informationen können im Rahmen des kontinuierlichen Verbesserungsprozesses Rückschlüsse auf die Relevanz von Kontextdokumenten für die Bearbeiter gezogen werden (Abschnitt 7.1.3).

Tabelle 8 fasst die Attribute des Aspekts „Lebenszyklus“ zusammen.

Tabelle 8: Attribute des Aspekts „Lebenszyklus“

Attributname	Bezogen auf	Datentyp	Wertemenge/Plausibilität
Exaktheit	Dokument	Aufzählungstyp	siehe Text
Ersteller	Kontext	Text	Wert muss gültigem Bearbeiternamen entsprechen.
Verantwortlicher	Kontext	Text	Wert muss gültigem Bearbeiternamen entsprechen.
Autor	Dokument	Text	keine Einschränkung
Gültigkeitszeitraum	Kontext	Feld [2] von Datum	Das durch die beiden Datumsangaben definierte Zeitintervall muss einem gültigen Zeitraum entsprechen.
Revisionsrhythmus	Kontext	Aufzählungstyp	siehe Text
Historie	Kontext	Menge von Operation	siehe Beschreibung des Attributs

6.3.4 Adressaten und Berechtigungen

Um die Menge der Bearbeiter angeben zu können, für die ein Kontextdokument relevant ist, und um Zugriffe auf Kontextdokumente und Änderungen der Attribute von Kontextdokumenten einschränken zu können, ist der Aspekt „Adressaten und Berechtigungen“ notwendig.

Zunächst soll der Begriff der Adressatengruppe diskutiert werden. Wie bereits in Abschnitt 6.2.2 angesprochen, ist eine weitere Strukturierung von Kontextdokumenten sinnvoll, indem zu jedem Kontextdokument die Adressatengruppe angegeben werden kann. Die Adressatengruppe bezeichnet dabei die Menge der Bearbeiter, für die das Kontextdokument relevant ist. So ist es z.B. möglich, dass in einem Unternehmen unterschiedliche Dokumente existieren, die Vorgaben bzgl. der Durchführung eines Prozesses enthalten, aber jeweils nur für einzelne Abteilungen gültig und damit relevant sind. Über die Angabe der entsprechenden Adressatengruppe zu jedem Dokument kann dieser Sachverhalt abgebildet werden. Der Benutzer kann

zur Spezifikation der Adressatengruppe eine Menge von Bearbeitern, Rollen (entspricht der Menge der Bearbeiter, die diese Rolle tragen) und Organisationseinheiten (entspricht der Menge der Bearbeiter, die dieser Organisationseinheit zugeordnet sind) angeben. Erfolgt keine explizite Angabe einer Adressatengruppe, ist das Kontextdokument uneingeschränkt für alle Bearbeiter relevant, und die Adressatengruppe enthält somit sämtliche Bearbeiter.

Der Zugriff auf Kontextdokumente soll in machen Fällen beschränkt sein. Daher werden im Folgenden verschiedene Arten von Berechtigungen erläutert. Um den Umfang der hier zu diskutierenden Berechtigungen zu verdeutlichen, sind zunächst die folgenden beiden Klassen von Operationen zu unterscheiden, auf die sich Berechtigungen im Umfeld von Kontextdokumenten grundsätzlich beziehen können:

- Operationen auf Dokumenten unabhängig von ihrer Nutzung in Kontexten:

Diese Klasse enthält Operationen auf Dokumenten, die vom Dokumententyp und der Informationsquelle, die das Dokument enthält, abhängig sind, wie z.B. das Lesen, Schreiben oder Löschen von Dokumenten, aber nicht spezifisch für die Nutzung von Dokumenten in Kontexten sind. Kontextdokumente können sich in unterschiedlichen Informationssystemen, wie z.B. Dateisystem, Groupware oder Dokumenten-Management-Systemen, befinden, die jeweils eigene Berechtigungen für ihren Dokumentenbestand verwalten. Es ist nicht das Ziel bei der Definition von Kontexten, ein allgemeines übergeordnetes Berechtigungssystem für Dokumente in beliebigen Informationssystemen zu erstellen. Vielmehr bleiben die Berechtigungskonzepte der einzelnen Informationssysteme auf der Ebene von Kontextdokumenten transparent. Fügt ein Bearbeiter ein Dokument in einen Kontext ein, muss der Bearbeiter durch das Berechtigungssystem des zugehörigen Informationssystems sicherstellen, dass die geeigneten Rechte auf dem Dokument definiert sind. Aus diesen Gründen werden die beschriebenen Operationen bei der Diskussion des Berechtigungsaspekts von Kontextdokumenten hier nicht erörtert.

- Kontextspezifische Operationen auf Dokumentbeschreibungen:

Diese Klasse von Operationen ist kontextspezifisch, und die einzelnen Operationen beziehen sich damit nicht auf das Dokument selbst, sondern auf die Bindung des Dokuments als Kontextdokument. Die entsprechenden Operationen und Berechtigungen werden in diesem Abschnitt diskutiert.

Die prinzipielle Idee von Kontexten liegt in einem kooperativen Aufbau und der kooperativen Bewahrung sekundären Wissens. Daraus folgt, dass einerseits sichergestellt werden muss, bestimmte Informationen nur eingeschränkten Bearbeitergruppen, wie z.B. dem Autor selbst oder einer bestimmten Organisationseinheit, zur Verfügung stellen zu können. Andererseits ist aber kein prinzipiell restriktiver Umgang mit den in Kontexten gespeicherten Informationen notwendig und sinnvoll, wenn ein kooperativer Wissensaufbau und ein verstärkter Wissenstransfer gefördert werden soll. Aus diesen Überlegungen heraus halten wir für die folgenden kontextspezifischen Operationen die Angabe von Berechtigungen für sinnvoll:

- Dokumentbeschreibung ändern:

Hier wird eine Menge von Bearbeitern, Rollen oder Organisationseinheiten angegeben, die die Attributwerte zu Kontextdokumenten modifizieren dürfen, also z.B. Sichtbarkeitsbedingungen, Gültigkeitszeiträume, Titel oder auch die Berechtigungen selbst. Ausgenommen ist lediglich das Attribut „Relevanz“, das von jedem Bearbeiter auch ohne besondere Berechtigung durch die Angabe einer zusätzlichen Einschätzung modifiziert werden darf.

- Kontextdokument aus Kontext entfernen:

Zu dieser Operation wird eine Menge von Bearbeitern, Rollen oder Organisationseinheiten spezifiziert, die das Dokument aus dem angegebenen Kontext entfernen darf. Es sei nochmals darauf hingewiesen, dass dabei das Dokument selbst nicht gelöscht wird, sondern es wird lediglich die Dokumentbeschreibung und somit die Referenz auf das in einem Informationssystem gespeicherte Dokument entfernt.

- Kontextdokument in der Dokumentenliste anzeigen:

Wie in Abschnitt 6.2.2 diskutiert, ist der Zugriff auf Kontexte grundsätzlich nicht beschränkt, um einen möglichst hohen Transfer von Wissen zu ermöglichen. Für einzelne Kontextdokumente kann allerdings die Sichtbarkeit des jeweiligen Kontextdokuments durch das Setzen des Attributs „Eingeschränkt sichtbar“ auf die Bearbeiter, Rollen oder Organisationseinheiten eingeschränkt werden, die durch das Attribut „Adressatengruppe“ spezifiziert wurden.

Tabelle 9 fasst die Attribute des Aspekts „Adressaten und Berechtigungen“ zusammen.

Tabelle 9: Attribute des Aspekts „Adressaten und Berechtigungen“

Attributname	Bezogen auf	Datentyp	Wertemenge/Plausibilität
Adressatengruppe	Kontext	Menge von Referenz	Jeder Wert muss einen Bearbeiter, eine Rolle, oder eine Organisationseinheit referenzieren.
Beschreibung ändern	Kontext	Menge von Referenz	Jeder Wert muss einen Bearbeiter, eine Rolle, oder eine Organisationseinheit referenzieren.
Dokument entfernen	Kontext	Menge von Referenz	Jeder Wert muss einen Bearbeiter, eine Rolle, oder eine Organisationseinheit referenzieren.
Eingeschränkt sichtbar	Kontext	Boolean	keine Einschränkung

6.3.5 Sichtbarkeit

Es ist in zahlreichen Fällen sinnvoll, die Bearbeitungssituation, für die ein Kontextdokument relevant ist, genauer festzulegen, als dies mit Hilfe der definierten Kontexte möglich ist. Beispielsweise sind bestimmte Kontextdokumente nur hilfreich, falls der aktuell zu bearbeitende Geschäftsfall eine hohe Priorität hat oder ein bei einer Vertragsverhandlung zu gewählender

Rabatt einen bestimmten Betrag überschreitet. Nur in diesem Fall sollen die entsprechenden Dokumente angezeigt werden. Diese Unterscheidung ist besonders wichtig, um den Bearbeiter nicht der Gefahr einer Überflutung mit Informationen auszusetzen, die aufgrund des Prozessstatus automatisch als nicht relevant erkannt werden könnten. Für das geschilderte Beispiel reicht also ein Geschäftsfall-Kontext, der alle relevanten Dokumente zu einem spezifischen Geschäftsfall zusammenfasst und anzeigt, nicht aus. Wir führen daher den Aspekt der Sichtbarkeit von Kontextdokumenten ein, um Dokumente in Abhängigkeit von bestimmten Bedingungen anzeigen zu können. In Bezug auf die Sichtbarkeitsbedingungen von Kontextdokumenten unterscheiden wir die folgenden Klassen (Abbildung 34).

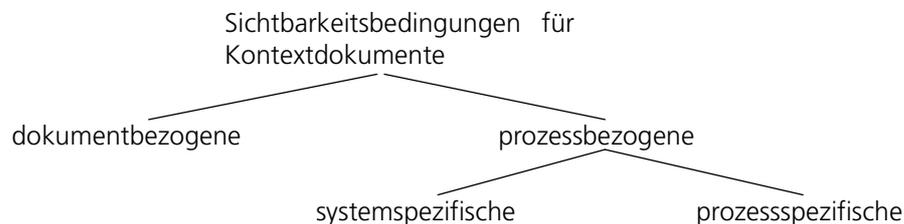


Abbildung 34: Klassen von Sichtbarkeitsbedingungen

Sichtbarkeitsbedingungen für Kontextdokumente können primär differenziert werden in dokumentbezogene und prozessbezogene Bedingungen. Dabei beziehen sich dokumentbezogene Bedingungen allein auf den Status von Dokumenten und sind unabhängig vom Status der Prozessbearbeitung. Als eine dokumentbezogene Sichtbarkeitsbedingung haben wir bei der Diskussion von Lebenszyklusinformationen (Abschnitt 6.3.3) das Attribut „Gültigkeitszeitraum“ identifiziert. Durch die Angabe eines Gültigkeitszeitraums wird die Sichtbarkeit eines Kontextdokuments (unabhängig von den Umständen der Prozessbearbeitung) eingeschränkt. Von besonderer Wichtigkeit sind bei wissensintensiven Prozessen prozessbezogene Sichtbarkeitsbedingungen, um unterschiedliche Funktionen von Kontextdokumenten (Abschnitt 5.2) abbilden zu können. In diesem Abschnitt werden wir daher diesen Aspekt genauer betrachten.

Prozessspezifische Bedingungen beziehen sich auf die „Workflow Relevant Data“ (Abschnitt 2.1.3), also auf die Prozessvariablen, die vom Modellierer im Prozessmodell festgelegt werden, während systemspezifische Bedingungen unabhängig von der Definition einzelner Prozesse sind und sich auf systemweit vorhandene Attribute z.B. von Geschäftsfällen beziehen. Ein Beispiel für eine prozessspezifische Bedingung zum Prozess „Contract Management“ ist „Discount > 10.000“, wobei „Discount“ eine im Prozessmodell definierte Prozessvariable darstellt. Beispiele für systemspezifische Bedingungen sind „Priorität des Geschäftsfalls = 1“ oder „Bearbeitungsfrist für Geschäftsfall überschritten = TRUE“, unter der Voraussetzung, dass das entsprechende WFMS die Attribute „Priorität“ und „Bearbeitungsfrist“ unterstützt. Im Folgenden bezeichnen wir sowohl Prozessvariablen als auch systemspezifische Attribute mit dem Begriff „Variable“.

Aufgrund dieser Überlegungen muss es für den Bearbeiter möglich sein, prozessbezogene Sichtbarkeitsbedingungen zu Kontextdokumenten angeben zu können. Die Formulierung dieser Bedingungen soll möglichst einfach sein, um das Konstrukt für den Bearbeiter überschaubar und verständlich zu halten. Wie oben beschrieben, sind prozessbezogene Sichtbarkeitsbedingungen teilweise systemspezifisch, und die Menge und das Format entsprechender systemspezifischer Variablen ist, wie die Untersuchung verschiedener WFMS zeigte, von System zu System sehr unterschiedlich. Daher soll im Folgenden keine Menge von Variablen als Grundlage für die Formulierung von Sichtbarkeitsbedingungen festgelegt werden, sondern die aufgeführten Variablen sind lediglich als Beispiele zu verstehen.

Zunächst sollen praxisrelevante systemspezifische Sichtbarkeitsbedingungen identifiziert werden: Das Ziel bei der Festlegung von Sichtbarkeitsbedingungen ist eine erweiterte kontextorientierte Unterstützung für die Bearbeitung von Prozessen. Diese Voraussetzung schränkt die Betrachtung auf die Workflow-Objekttypen ein, die zur Laufzeit von Prozessen dynamischen Änderungen unterworfen sind. Damit sind z.B. Workflow-Objekttypen wie „Prozess“ für die weitere Untersuchung nicht von Interesse, denn die zugehörigen Variablen - typische Beispiele sind „Name“, „Beschreibung“, „Kategorie“ oder „Status“ (freigegeben, nicht freigegeben) - werden normalerweise zur Laufzeit nicht geändert. Aufbauend auf der Untersuchung von WFMS halten wir die beiden Workflow-Objekttypen „Geschäftsfall“ und „Aktivitätsausführung“ (im Unterschied zur Aktivität bezeichnet die Aktivitätsausführung die Durchführung einer Aktivität in einem konkreten Geschäftsfall) als geeignete Basis zur Formulierung von relevanten Sichtbarkeitsbedingungen.

In Bezug auf die prozessspezifischen Bedingungen, die sich jeweils auf Prozessvariablen beziehen, machen wir über ein zu unterstützendes WFMS folgende Annahme: Prozessvariablen werden zu einem Prozess insgesamt definiert, d.h. der Gültigkeitsbereich der Variablen ist der gesamte Prozess ohne Einschränkung auf einzelne Aktivitäten.

Zusammenfassend sollen also auf der Basis von Variablen des Workflow-Objekttyps „Geschäftsfall“ sowohl systemspezifische als auch prozessspezifische Sichtbarkeitsbedingungen möglich sein, auf der Basis von Variablen des Workflow-Objekttyps „Aktivitätsausführung“ sollen systemspezifische Sichtbarkeitsbedingungen ermöglicht werden. Im Folgenden geben wir beispielhaft einige Variablen und mögliche zugehörige Bedingungen an.

Geschäftsfall:

- systemspezifische Variablen und Bedingungen

Name	(Name enthält „Fa. Siemens“; Informationen zu einem speziellen Kunden)
Beschreibung	(Beschreibung enthält „Grosskundendiscount“)
Erstellungszeitpunkt	(Erstellungszeitpunkt \geq „1.3.2000“; Ab diesem Datum gelten neue Bearbeitungsrichtlinien)
Priorität	(Priorität = „hoch“)
Bearbeitungsfrist	(Bearbeitungsfrist_überschritten = TRUE; Es ist eine Ausnahme aufgetreten, für die weitere Informationen benötigt werden)
Status	(Status = „Geschäftsfall suspendiert“)

- prozessspezifische Variablen und Bedingungen
(hier beispielhaft für den Prozess „Contract Management“)

Discount	(Discount \geq 10.000; Es wird ein hoher Rabatt gewährt. Hierbei müssen besondere Genehmigungsverfahren beachtet werden.)
Kundenrating	(Kundenrating = „A-Kunde“)

Aktivitätsausführung:

- systemspezifische Variablen und Bedingungen

Deadline	(Bearbeitungsfrist_überschritten = TRUE)
Status	(Status = „Aktivität ist zugewiesen“)

Wir beschreiben im Weiteren, in welchen Kontexten die Angabe von Sichtbarkeitsbedingungen zu Kontextdokumenten sinnvoll und zulässig ist (dies wird in Tabelle 10 zusammengefasst; als Grundlage dient die Darstellung der Kontexttypen in Tabelle 4). Kontexttypen, die an ein kontextbildendes Objekt der Organisationsstruktur gebunden sind (und damit Dokumente enthalten, die sich nicht auf einen konkreten Prozess beziehen), enthalten keine Möglichkeiten zur Spezifikation von Sichtbarkeitsbedingungen, da sich diese auf Variablen von Geschäftsfällen oder Aktivitätsausführungen und damit immer auf einen konkreten Prozess beziehen müssen. Ebenso können in Kontexten, die mit Hilfe der kontextbildenden Objekttypen „Dokumentvorlage“ und „Werkzeug“ gebildet werden, keine Sichtbarkeitsbedingungen angegeben werden. Da entsprechend der in Abbildung 32 vorgestellten Menge kontextbildender Objekttypen sowohl „Werkzeug“ als auch „Dokumentvorlage“ in einer many-to-many-Relation zu „Prozess“ steht, sollen entsprechende Kontexte Dokumente enthalten, die nicht für bestimmte Prozesse relevant sind, sondern allgemeine Hilfen, Fehlermeldungen, etc. zu Werkzeugen bzw. Dokumentvorlagen bereitstellen.

Für die jeweils durch die kontextbildenden Objekttypen „Prozess“ und „Aktivitätsblock“ gebildeten Kontexttypen soll es möglich sein, Sichtbarkeitsbedingungen, die sich auf Geschäftsfallvariablen (sowohl system- als auch prozessspezifische) beziehen, angeben zu können. Bei Kontexten, die unabhängig von konkreten Geschäftsfällen sind, wird dann zur Laufzeit

geprüft, ob die Sichtbarkeitsbedingung von dem Geschäftsfall erfüllt wird, den der Bearbeiter gerade durchführt (in der Tabelle mit $C_{GF}(GF_{akt})$ beschrieben - die Bedingung C_{GF} wird also auf den aktuellen Geschäftsfall GF_{akt} angewandt; s.u. Beispiel 1). Ist der Kontext abhängig von einem konkreten Geschäftsfall (rechte Tabellenspalte), wird die Sichtbarkeit von Kontextdokumenten abhängig von den Variablenwerten dieses konkreten Geschäftsfalls gemacht (in der Tabelle mit $C_{GF}(GF)$ beschrieben; s.u. Beispiel 2).

Bei Aktivitäts-Kontexttypen sind sowohl die beschriebenen Sichtbarkeitsbedingungen zu Geschäftsfällen als auch Sichtbarkeitsbedingungen zur Aktivitätsausführung verfügbar. Die Bedingungen beziehen sich immer auf die Ausführung der speziellen Aktivität, an die der Kontext gebunden ist. Wie im Fall des Prozesses gilt auch hier: Ist der Kontext unabhängig von konkreten Geschäftsfällen, wird die Bedingung zur Laufzeit abhängig vom Status der Aktivitätsausführung im gerade bearbeiteten Geschäftsfall geprüft (in der Tabelle durch $C_A(GF_{akt}, A)$ beschrieben). Bezieht sich der Kontext dagegen nicht nur auf eine konkrete Aktivität, sondern auch auf einen konkreten Geschäftsfall, bilden die Variablenwerte dieser beiden Objekte die Basis der Prüfung der Bedingung (in der Tabelle durch $C_A(GF, A)$ beschrieben). Im Folgenden werden zur Veranschaulichung zwei Beispiele aufgeführt.

Tabelle 10: Mögliche Sichtbarkeitsbedingungen für Kontextdokumente in unterschiedlichen Kontexttypen

Kontextbildende Objekttypen	unabhängig von konkreten Geschäftsfällen	Geschäftsfall (GF)
Prozess	$C_{GF}(GF_{akt})$	$C_{GF}(GF)$
Aktivität (A)	$C_{GF}(GF_{akt})$ und $C_A(GF_{akt}, A)$	$C_{GF}(GF)$ und $C_A(GF, A)$
Aktivitätsblock	$C_{GF}(GF_{akt})$	Kontexte nicht existent (vgl. Tabelle 4)
Dokumentvorlage	keine Sichtbarkeitsbedingungen möglich	Kontexte nicht existent (vgl. Tabelle 4)
Werkzeug	keine Sichtbarkeitsbedingungen möglich	Kontexte nicht existent (vgl. Tabelle 4)
Organisationseinheit	keine Sichtbarkeitsbedingungen möglich	Kontexte nicht existent (vgl. Tabelle 4)
Rolle	keine Sichtbarkeitsbedingungen möglich	Kontexte nicht existent (vgl. Tabelle 4)
Bearbeiter	keine Sichtbarkeitsbedingungen möglich	Kontexte nicht existent (vgl. Tabelle 4)

Beispiel 1:

Im Kontext zum Prozess „Contract Management“ wird ein Dokument mit zusätzlichen Anweisungen abgelegt, dass nur sichtbar sein soll, wenn der Discountbetrag größer als 10.000 ist. Führt der Bearbeiter einen Geschäftsfall durch, für den diese Bedingung erfüllt ist, und greift auf den Kontext zu, ist das entsprechende Dokument sichtbar.

Beispiel 2:

Im durch die kontextbildenden Objekte „Contract Management“ (Prozess) und „Comdirect Bank“ (Geschäftsfall) festgelegten Kontext fügt der Verkäufer ein Dokument mit Hinweisen ein, wie in diesem speziellen Geschäftsfall zu verfahren ist, sollte die Bearbeitungsfrist überschritten werden. Tritt später diese Ausnahme ein, ist das Dokument in diesem Kontext sichtbar.

Die Spezifikation von Sichtbarkeitsbedingungen zu Kontextdokumenten soll nach folgenden Regeln ermöglicht werden: Eine Sichtbarkeitsbedingung kann aus mehreren Teilbedingungen bestehen, die logisch disjunktiv oder konjunktiv verknüpft werden. Eine Teilbedingung ist nach dem Schema <Variable> <Operator> <Konstante> aufgebaut, wobei die Variable sowohl den Datentyp der Konstanten als auch die Menge der zulässigen Operatoren festlegt. Tabelle 11 zeigt die in Teilbedingungen möglichen Datentypen von Variablen und dazu die jeweils zulässigen Operatoren und Konstanten. Dadurch dass die Zahl der Datentypen beschränkt ist und auf der rechten Seite der Bedingung nur Konstanten zugelassen werden, sollen Sichtbarkeitsbedingungen bewusst einfach definiert werden können und damit für den Bearbeiter beherrschbar sein.

Tabelle 11: Zulässige Datentypen, Operatoren und Konstanten in Teilbedingungen

Datentyp	Operatoren	Konstante
Boolean	=, <>	TRUE FALSE
Integer	<, >, =, <>	IntegerKonstante
Real	<, >, =, <>	RealKonstante
Datum	<, >, =, <>	DateKonstante
Text	=, <>, Enthält	TextKonstante
Aufzählung	<, >, =, <>	AufzählungKonstante

WFMS verwalten im Allgemeinen eine Vielzahl von Variablen, um den Status eines Geschäftsfalls bzw. einer Aktivitätsausführung zu charakterisieren. Ebenso sind zu einem Prozess oftmals eine Vielzahl von Prozessvariablen definiert. Von dieser Menge an Variablen ist nur ein geringer Teil zur Definition von Sichtbarkeitsbedingungen geeignet. Um nun dem Bearbeiter nicht eine unüberschaubare Menge von Variablen bei der Spezifikation von Sichtbarkeitsbedingungen anzubieten, ist sowohl bei der Konfiguration von Kontexten für ein bestimmtes WFMS als auch bei der Definition jedes einzelnen Prozesses die Auswahl einer Menge von jeweils sinnvollen Variablen durch den Modellierer notwendig, die als Grundlage für system-spezifische bzw. prozessspezifische Sichtbarkeitsbedingungen dienen kann.

Tabelle 12 fasst die Attribute des Aspekts „Sichtbarkeit“ zusammen. Wird keine Sichtbarkeitsbedingung angegeben, ist das Dokument stets sichtbar.

Tabelle 12: Attribute des Aspekts „Sichtbarkeit“

Attributname	Bezogen auf	Datentyp	Wertemenge/Plausibilität
Sichtbarkeit	Kontext	Bedingung	siehe Text

6.3.6 Bereitstellung

Die Bereitstellung der richtigen Information zum richtigen Zeitpunkt kann auf grundsätzlich zwei verschiedene Arten erfolgen. Wird der Bearbeiter selbst aktiv und greift auf Dokumente zu, wird dieser Bereitstellungsmodus als „Pull-Prinzip“ bezeichnet. Ist dagegen das System aktiv und stellt dem Bearbeiter ohne dessen aktive Teilnahme Dokumente zu, spricht man von „Push-Prinzip“. WFMS, die einen Teil des primären Wissens in Form von Prozessmodell dokumenten abbilden und gemäß dem Prozessmodell an den Bearbeiter übermitteln, folgen dem Push-Prinzip. Die Bereitstellung von sekundärem Wissen durch Kontextdokumente soll gemäß sowohl dem Push- als auch dem Pull-Prinzip ermöglicht werden.

Die aktive Bereitstellung von Informationen nach dem Push-Ansatz sollte stets das Ziel verfolgen, die Relevanz der Information für potentielle Adressaten zu bewerten und die Informationen den Adressaten zum Zeitpunkt des Informationsbedarfs zuzustellen (vgl. diesbezüglich die Charakterisierung informationslogistischer Anwendungen [Webe99] und von Gruppenwahrnehmungskontexten [GP00]). Während die Auswahl von Adressaten, die an einer Information interessiert sind, über Interessensprofile von Benutzern für viele Anwendungsgebiete möglich ist und in vielen Wissensmanagementsystemen bereits realisiert ist, ist die Frage des korrekten Zeitpunkts der Informationsübermittlung weitaus schwieriger zu beantworten. Viele Systeme bieten daher lediglich Push-Kanäle an, die eine Information, wie z.B. ein Dokument, entweder direkt nach der Erstellung oder in Abhängigkeit von zeitlichen Ereignissen an eine aufgrund von Interessensprofilen ausgewählte Menge von Benutzern senden. Da dies im schlimmsten Fall zu einer Überflutung der Benutzer mit Informationen zum falschen Zeitpunkt führt, ist der Erfolg dieser Ansätze für viele Anwendungsgebiete zweifelhaft.

Für den hier zu diskutierenden Fall einer aktiven Bereitstellung von Kontextdokumenten sind zwei Prinzipien für die weitere konzeptionelle Ausarbeitung elementar: Sowohl die Auswahl der Adressaten als auch der Zeitpunkt der Bereitstellung eines Kontextdokuments soll sich erstens am konkreten Wissensbedarf der Adressaten in einer Bearbeitungssituation orientieren. Zweitens muss die Spezifikation einer aktiven Bereitstellung von Kontextdokumenten möglichst einfach für den Bearbeiter sein, um die Praxistauglichkeit des Ansatzes zu gewährleisten. Dazu sind die folgenden Aspekte zu untersuchen:

- Bereitstellungsmodus:

Die Bereitstellung von Kontextdokumenten soll sowohl nach dem Push- als auch nach dem Pull-Ansatz möglich sein. Im Fall der Bereitstellung per Pull-Prinzip sind keine weiteren Attribute zur Beschreibung von Kontextdokumenten notwendig. Kontextdokumente werden per Default nach dem Pull-Prinzip bereitgestellt.

Die folgenden Aspekte sind lediglich für den Fall der Bereitstellung von Kontextdokumenten per Push-Prinzip relevant:

- Zeitpunkt der Bereitstellung:
Wann sollen Kontextdokumente zugestellt werden? Wir werden diesbezüglich ein ereignisbasierten Konzept vorschlagen.
- Adressatengruppe:
Welchen Bearbeitern werden Kontextdokumente aktiv durch das System bereitgestellt?
- Wiederholbarkeit:
Wie oft wird ein Kontextdokument übermittelt? Im Zusammenhang mit einem ereignisbasierten Konzept kann hier unterschieden werden, ob ein Kontextdokument stets übermittelt wird, falls ein definiertes Ereignis eintritt, oder ob das Kontextdokument nur einmal an einen Bearbeiter versandt wird, um eine Informationsüberflutung zu vermeiden.

Die Spezifikation einer aktiven Bereitstellung von Kontextdokumenten soll ereignisbasiert nach dem Event-Condition-Action-Paradigma erfolgen. Dabei wird die Aktion, also die aktive Bereitstellung des Kontextdokuments, durch ein Ereignis ausgelöst, welches eine an das Ereignis gekoppelte Bedingung erfüllen muss. Es sind zunächst Ereignisse zu identifizieren, die während der Durchführung von Prozessen auftreten und, da sie direkt mit Aktionen des Bearbeiters in Beziehung stehen, veränderte Wissensbedarfe des Bearbeiters bewirken können. Tabelle 13 stellt einen Vorschlag für eine Menge von Ereignissen vor, die diesbezüglich relevant sind. Es sei hier nochmals darauf hingewiesen, dass Kontextdokumente eine Unterstützung bei der Durchführung von Prozessen leisten sollen, nicht aber die Phasen der Erhebung oder Modellierung unterstützen. Daher sind Ereignisse, die sich auf den Lebenszyklus von Prozessen beziehen, wie z.B. „Prozess erstellt“ oder „Prozess freigegeben“, hier nicht von Interesse. Die dargestellten Ereignisse wurden nicht von einem konkreten WFMS oder einem abstraktem Beschreibungsansatz wie dem der WfMC abgeleitet. Vielmehr handelt es sich um eine Möglichkeit der Darstellung, die aus einer praxisorientierten Sichtweise heraus gewählt wurde, da die Menge der aufgeführten Ereignisse größtenteils von konkreten WFMS unterstützt werden.

Ist ein Ereignis ausgewählt, ist damit bereits die Aktion vorgegeben, nämlich die aktive Übermittlung des Kontextdokuments an den Bearbeiter, der mit dem Ereignis in Verbindung steht bzw. das Ereignis ausgelöst hat. Komplexer ist allerdings die Definition der Bedingung, die das Ereignis einschränkt. Diese soll einerseits die Menge der Workflow-Objekte, auf die sich das Ereignis beziehen kann, einschränken und andererseits die Menge der Bearbeiter begrenzen, die das Ereignis auslösen können. Im in Abschnitt 4.2 vorgestellten Beispiel soll dem Verkäufer eine Änderung der Richtlinien zu Rabattierungen bei der Vorbereitung auf den Termin mit einem Kunden zugestellt werden. Nehmen wir weiter an, dass sich diese Richtlinie lediglich auf einen bestimmten Verkaufsdistrikt bezieht, dann kann die Spezifikation der aktiven Bereitstellung des entsprechenden Dokuments gemäß des Event-Condition-Action-Ansatzes folgendermaßen erfolgen: Als Ereignis wird „Aktivität gestartet“ ausgewählt, wodurch festgelegt ist, dass dem Bearbeiter, der dieses Ereignis auslöst, automatisch das Kontextdokument zugestellt wird. Die zugehörige Bedingung soll nun sowohl

- die Ereignismenge beschränken (es sind nur Ereignisse relevant, die den Start der Aktivität „Rabattierung mit Kunden verhandeln“ repräsentieren) als auch

- die Menge der Bearbeiter, die das Ereignis auslösen können und denen daraufhin das entsprechende Dokument zugestellt werden soll, einschränken (Bearbeiter ist Mitarbeiter im „Verkaufsdistrikt Central“).

Tabelle 13: Ereignisse zur aktiven Bereitstellung von Kontextdokumenten

Ereignisklasse	Ereignis	Beschreibung
Bearbeiter (E_B)	eingeloggt	Ein Bearbeiter meldet sich beim System an.
	ausgeloggt	Ein Bearbeiter meldet sich beim System ab.
Geschäftsfall (E_{GF})	gestartet	Ein Bearbeiter instanziiert einen Geschäftsfall.
	suspendiert	Ein Bearbeiter suspendiert einen Geschäftsfall.
	wiederaufgenommen	Ein Bearbeiter nimmt einen Geschäftsfall nach Suspendierung wieder auf.
	beendet	Ein Bearbeiter beendet einen Geschäftsfall, d.h. er beendet die letzte aktive Aktivität.
Aktivität (E_A)	abgebrochen	Ein Bearbeiter bricht einen Geschäftsfall ab.
	zugewiesen	Eine Aktivität wird einem Bearbeiter zugewiesen.
	gestartet	Ein Bearbeiter startet eine Aktivität.
Dokument (E_D)	beendet	Ein Bearbeiter beendet eine Aktivität.
	erzeugt	Ein Bearbeiter erzeugt ein neues Dokument.
Werkzeug (E_W)	geöffnet	Ein Bearbeiter öffnet ein Dokument.
	gestartet	Ein Bearbeiter startet ein Werkzeug.

Da das Einfügen von Kontextdokumenten in Kontexte von einem Bearbeiter integriert in seine eigentliche Aufgabe ohne hohe zusätzliche zeitliche Aufwände erledigt werden sollte, ist eine einfache und intuitive Spezifikation der Bedingungen notwendig.

Eine Lösung bietet die Spezifikation der aktiven Bereitstellung eines Kontextdokuments auf der Basis der Funktion eines konkreten Kontexts und unter Berücksichtigung der Adressatengruppe des Kontextdokuments (Abschnitt 6.3.4): Da Kontexte jeweils eine bestimmte Bearbeitungssituation unterstützen, kann für ein Kontextdokument innerhalb eines konkreten Kontexts einerseits die Menge der möglichen Ereignisse automatisch auf eine sinnvolle Untermenge eingeschränkt werden, und andererseits ist die zugehörige Bedingung durch den Kontext und die Spezifikation der Adressatengruppe des zugehörigen Kontextdokuments implizit vorgegeben. Ist z.B. ein Dokument für die Aktivität „Rabattanfrage erstellen“ relevant, wird es von einem Bearbeiter in den zu dieser Aktivität zugehörigen Kontext eingefügt. Soll das Kontextdokument aktiv bereitgestellt werden, ist es daher sinnvoll, dass dem Bearbeiter Ereignisse der Ereignisklasse „Aktivität“ (E_A) angeboten werden, nicht aber z.B. der Klasse „Geschäftsfall“ (E_{GF}), da letztere keinen Bezug zu der konkreten Aktivitätsdurchführung und damit zu der durch den Kontext vorgegebenen Bearbeitungssituation besitzen. Wählt der Bearbeiter ein Ereignis, wie z.B. „Aktivität gestartet“, aus, ist erstens die Menge der Workflow-Objekte, auf

die sich das Ereignis bezieht, durch die Funktion des Kontexts vorgegeben (nämlich: Aktivität = „Rabattanfrage erstellen“). Zweitens wird die Menge der Bearbeiter, die das Ereignis auslösen können und denen daraufhin das Kontextdokument aktiv bereitgestellt werden soll, durch Rückgriff auf die Adressatengruppe des Kontextdokuments eingeschränkt: Nur wenn der Bearbeiter, der das Ereignis auslöst, Mitglied der Adressatengruppe ist, ist dieser Teil der Bedingung erfüllt. Auch die Aktion muss, wie oben bereits erwähnt, nicht weiter spezifiziert werden: Es wird dem Bearbeiter das Kontextdokument aktiv bereitgestellt, der das Ereignis in einem beliebigen Geschäftsfall, auslöst.

Im Folgenden sollen nun für die existierenden Kontexttypen (vgl. Tabelle 4) die den Bearbeitern zur Auswahl verfügbaren Ereignisse und die damit implizit vorgegebenen Bedingungen vorgestellt werden (Tabelle 14). Zu jedem Kontexttyp werden die verfügbaren Ereignisklassen angegeben, und zu jeder Ereignisklasse wird die implizit vorgegebene Bedingung in der Form „Ereignisklasse: Bedingung“ dargestellt. Dabei bezeichnet B_{akt} den aktuellen Bearbeiter, der ein Ereignis auslöst, GF_{akt} den aktuellen Geschäftsfall, in dem ein Ereignis ausgelöst wird, A_{akt} die aktuelle Aktivität, zu der ein Ereignis ausgelöst wird, und D_{Adr} die Adressatengruppe (Abschnitt 6.3.4) des Kontextdokuments, auf das sich die Spezifikation der aktiven Bereitstellung bezieht.

Die folgenden Ereignisse sollen in den jeweiligen Kontexttypen für den Bearbeiter auswählbar sein: In Prozess-Kontexten sind Ereignisse der Klasse „Geschäftsfall“ (E_{GF}) vorzusehen. In Aktivitäts-Kontexten und Aktivitätsblock-Kontexten sind Ereignisse der Klasse „Aktivität“ (E_A) auswählbar, entsprechend in Dokumentvorlagen-Kontexten Ereignisse der Klasse „Dokument“ (E_D) und in Werkzeug-Kontexten Ereignisse der Klasse „Werkzeug“ (E_W). Schließlich sind in Kontexten, die an ein kontextbildendes Objekte der Organisationsstruktur gebunden sind, Ereignisse der Klasse „Bearbeiter“ (E_B) auswählbar. Die zugehörigen Bedingungen enthalten stets, wie oben erläutert, eine Teilbedingung zur Einschränkung der Menge der Bearbeiter, die das Ereignis auslösen können, über die Adressatengruppe des Kontextdokuments (B_{akt} IN D_{Adr}). Daneben existiert eine kontexttypspezifische Teilbedingung: Für einen Prozess-Kontexttyp lautet diese bspw., dass der aktuelle Geschäftsfall, in dem das Ereignis ausgelöst wurde, zu dem Prozess gehören muss, auf den sich der entsprechende Kontext bezieht (GF_{akt} IN P), und für einen Aktivitäts-Kontexttyp legt die Teilbedingung fest, dass die aktuelle Aktivität, für die das Ereignis festgelegt wurde, der Aktivität entspricht, an die der entsprechende Kontext gebunden ist. Sind die entsprechenden Kontexttypen abhängig von einem konkreten Geschäftsfall (rechte Spalte der Tabelle), werden die einzelnen Bedingungen um den Ausdruck $GF_{akt} = GF$ erweitert. Dieser Ausdruck gibt an, dass der aktuelle Geschäftsfall, in dem das Ereignis ausgelöst wurde, dem Geschäftsfall entsprechen muss, an den der Kontext gebunden ist.

Tabelle 14: Ereignisse und Bedingungen zur aktiven Bereitstellung von Kontextdokumenten

Kontextbildende Objekttypen	unabhängig von konkreten Geschäftsfällen	Geschäftsfall (GF)
Prozess (P)	$E_{GF}: GF_{akt} \text{ IN } P \wedge B_{akt} \text{ IN } D_{Adr}$	$E_{GF}: GF_{akt} = GF \wedge B_{akt} \text{ IN } D_{Adr}$
Aktivität (A)	$E_A: A_{akt} = A \wedge B_{akt} \text{ IN } D_{Adr}$	$E_A: GF_{akt} = GF \wedge A_{akt} = A \wedge B_{akt} \text{ IN } D_{Adr}$
Aktivitätsblock (AB)	$E_A: A_{akt} \text{ IN } AB \wedge B_{akt} \text{ IN } D_{Adr}$	Kontexte nicht existent (vgl. Tabelle 4)
Dokumentvorlage (DV)	$E_D: D_{akt} \text{ IN } DV \wedge B_{akt} \text{ IN } D_{Adr}$	Kontexte nicht existent (vgl. Tabelle 4)
Werkzeug (W)	$E_W: W_{akt} = W \wedge B_{akt} \text{ IN } D_{Adr}$	Kontexte nicht existent (vgl. Tabelle 4)
Organisationseinheit (OE)	$E_B: B_{akt} \text{ IN } D_{Adr}$	Kontexte nicht existent (vgl. Tabelle 4)
Rolle (R)	$E_B: B_{akt} \text{ IN } D_{Adr}$	Kontexte nicht existent (vgl. Tabelle 4)
Bearbeiter (B)	$E_B: B_{akt} \text{ IN } D_{Adr}$	Kontexte nicht existent (vgl. Tabelle 4)

Aus der obigen Darstellung folgt: Bei der Spezifikation der aktiven Bereitstellung eines Kontextdokuments muss vom Bearbeiter lediglich das zugehörige Ereignis aus einer vorgegebenen Menge ausgewählt werden. Sowohl die zugehörige Bedingung als auch die Aktion sind implizit bereits durch die Auswahl des Kontexts und die Adressatengruppe des Kontextdokuments vorgegeben. Damit folgt das vorgestellte Konzept den beiden in diesem Abschnitt angegebenen Zielen, erstens der Bereitstellung von Dokumenten zum Zeitpunkt des erwarteten Wissensbedarfs und zweitens der unkomplizierten und praxistausgleichlichen Spezifikation der aktiven Bereitstellung durch den Bearbeiter.

Ereignisse sind zeitlos. Daher ist eine konjunktive Verknüpfung von Ereignissen zu einem Kontextdokument nicht zulässig. Die disjunktive Verknüpfung von Ereignissen, die zu einer aktiven Zustellung eines Kontextobjekts führen, soll dagegen möglich sein: Werden mehrere Ereignisse ausgewählt, sind diese automatisch disjunktiv verbunden.

Bisher nicht betrachtet wurde der eingangs geforderte Aspekt der Wiederholbarkeit einer aktiven Bereitstellung. Unabhängig von der gewählten Ereignismenge sollen zwei unterschiedliche Wiederholbarkeitsmodi auswählbar sein:

- einmalig:

Die Zustellung des Kontextdokuments erfolgt an den entsprechenden Bearbeiter, wenn das entsprechende Ereignis eintritt, die Bedingungen erfüllt sind und dem Bearbeiter das Dokument in der aktuellen Version noch nicht aktiv zugestellt wurde.

- unbedingt:

Die Zustellung des Kontextdokuments erfolgt stets, wenn das entsprechende Ereignis eintritt und die Bedingungen erfüllt sind.

Abbildung 35 zeigt zusammenfassend die einzelnen Bedingungen, die zur aktiven Bereitstellung eines Kontextdokuments geprüft werden müssen.

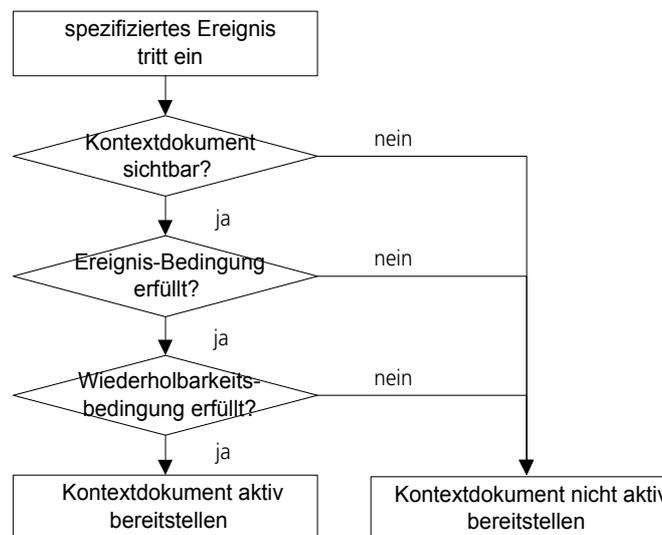


Abbildung 35: Aktive Bereitstellung eines Kontextdokuments

Im Folgenden werden zwei Beispiele aufgeführt, an denen die oben geschilderten Aspekte zur aktiven Bereitstellung von Kontextdokumenten veranschaulicht werden:

Beispiel 1:

In diesem Abschnitt wurde bereits das Beispiel aufgeführt, dass einem Verkäufer des Verkaufsdistrikts „Central“ ein geändertes Dokument bei der Vorbereitung auf einen Kundentermin zugestellt werden muss. Eine Spezifikation des Kontextdokuments auf der Basis dieser Anforderungsdefinition würde folgendermaßen aussehen:

- Das Dokument wird in den Kontext eingefügt, der an die Aktivität „Rabattierung mit Kunden verhandeln“ gebunden ist und unabhängig von konkreten Geschäftsfällen ist.
- Im Aspekt „Adressaten und Berechtigungen“ wird die Adressatengruppe auf die Organisationseinheit „Verkaufsdistrikt Central“ eingeschränkt.
- Im Aspekt „Bereitstellung“ wird das Ereignis „Aktivität gestartet“ ausgewählt und die Wiederholbarkeit auf „einmalig“ festgelegt.

Beispiel 2:

Wie bereits in Kapitel 5 erwähnt, sollte eine Unterstützung wissensintensiver Prozesse auch die aktive Abfrage von Wissen, wie z.B. Erfahrungen der Bearbeiter, durch das System umfassen. Daher soll in diesem Beispiel jeder Benutzer angeleitet werden, seine Erfahrungen bei der Bearbeitung der Aktivität „Rabattanfrage erstellen“ abzugeben, falls der dem Kunden zu gewährende Rabatt größer als ein bestimmter Betrag ist und es sich daher um eine komplexe Rabattierung handelt, die in späteren Geschäftsfällen interessant sein kann.

- Der Modellierer fügt ein leeres Dokument mit dem Titel „Erfahrungen mit der Anwendbarkeit der aktuellen Rabattstruktur bei hohen Rabatten“ in den Kontext ein, der durch die Aktivität „Rabattanfrage erstellen“ festgelegt ist und unabhängig konkreten Geschäftsfällen ist.
- Im Aspekt „Sichtbarkeit“ wird die Bedingung („Discount \geq 10.000“) spezifiziert.
- Im Aspekt „Bereitstellung“ wird das Ereignis „Aktivität beendet“ ausgewählt und die Wiederholbarkeit auf „unbedingt“ festgelegt.

Die beschriebenen Attribute stellen sicher, dass das Dokument jedem Bearbeiter (es wurde keine Einschränkung der Adressatengruppe festgelegt) nach der Durchführung der Aktivität „Rabattanfrage erstellen“ aktiv angezeigt wird, falls der zugehörige Geschäftsfall die Sichtbarkeitsbedingung erfüllt. Das System leitet damit die Bearbeiter ohne Zwang an, ihre Erfahrungen in einem Kontextdokument zu explizieren.

Tabelle 15 fasst die Attribute des Aspekts „Bereitstellung“ zusammen.

Tabelle 15: Attribute des Aspekts „Bereitstellung“

Attributname	Bezogen auf	Datentyp	Wertemenge/Plausibilität
Ereignisse	Kontext	Menge von Referenz	Jeder Wert muss ein Ereignis referenzieren.
Wiederholbarkeit	Kontext	Aufzählungstyp	[einmalig, unbedingt]

6.4 Eine Standardbibliothek für Informationen aus Workflow-Management-Systemen

In Abschnitt 5.1 haben wir bereits erläutert, dass zu einer Verwaltung des gesamten prozessbezogenen Wissens insbesondere die Integration der vom WFMS verwalteten Prozessmodell- und Laufzeitinformationen (Abbildung 25) erforderlich ist. Diese Informationen über z.B. laufende Fälle und ihre Eigenschaften oder die Personen, die bestimmte Aktivitäten besonders häufig durchführen, können zur Entscheidungsfindung oder als allgemeine Hintergrundinformationen für den Bearbeiter relevant sein und sollten in Form von Kontextdokumenten verfügbar sein. Dieser Typ von Kontextdokumenten wird im Folgenden als Systemdokumente bezeichnet (siehe die Auflistung der Standardtypen von Kontextdokumenten in Tabelle 6).

Die Menge der jeweils für den Bearbeiter relevanten Informationen aus dem WFMS ist zwar von der konkreten Bearbeitungssituation abhängig, es erscheint aber sinnvoll, einen initialen Bestand von Informationen zu standardisieren und in einer Standardbibliothek von Dokumenttypen bereitzustellen. Die Elemente dieser Standardbibliothek wurden dabei so gewählt, dass sie Informationen enthalten, die dem Bearbeiter als wichtige Hintergrundinformationen oder Entscheidungsunterstützung dienen können. Dabei wurden sowohl Informationen zu den einzelnen kontextbildenden Workflow-Objekten als auch historische Daten¹, die bei der Ausführung von Geschäftsfällen durch WFMS anfallen, in Betracht gezogen. Die Instanzen der im Weiteren beschriebenen Dokumenttypen, also die Systemdokumente, greifen bei jedem Öffnen des Dokuments durch den Bearbeiter auf den Informationsbestand des WFMS zu und enthalten daher stets die aktuellen Informationen. Wird also ein Systemdokument vom Typ „Informationen zu einem Prozess“ instanziiert, muss initial der entsprechende Prozess spezifiziert werden. Das Dokument enthält daraufhin stets die aktuellen Informationen zum angegebenen Prozess, auch wenn die Prozessdefinition modifiziert wird. Die Instanziierung von Systemdokumenttypen, die daraufhin als Systemdokumente in entsprechende Kontexte eingefügt werden, soll sowohl zum Modellierungszeitpunkt durch den Modellierer als auch zur Laufzeit durch den Bearbeiter möglich sein. Beim Aufbau der Standardbibliothek von Systemdokumenttypen unterscheiden wir drei Komplexitätsstufen der bereitgestellten Informationen.

Komplexitätsstufe 1: Statische Informationen, die aus Prozessmodellen oder dem Organisationsmodell abgeleitet sind

Systemdokumente dieser Klasse enthalten Informationen, die sich direkt aus den vom WFMS verwalteten Prozessmodellen bzw. dem Organisationsmodell ableiten lassen und dem Bearbeiter als Hintergrundinformation zur Verfügung stehen sollten. Die folgenden Informationen werden als Dokumenttypen in der Standardbibliothek bereitgestellt:

1. Informationen zu einem Prozess
(Parameter: Prozess)
2. Informationen zu einer Aktivität
(Parameter: Aktivität)
3. Informationen zu einer Dokumentvorlage
(Parameter: Dokumentvorlage)
4. Informationen zu einem Werkzeug
(Parameter: Werkzeug)
5. Informationen zu einem Bearbeiter
(Parameter: Bearbeiter)
6. Informationen zu einer Rolle
(Parameter: Rolle)

1. Nur ein geringer Teil der historischen Daten eines WFMS sind für die hier betrachtete Aufgabe der Unterstützung des Bearbeiters zur Laufzeit relevant. Vgl. [Hahn99] für eine vollständige Übersicht des historischen Aspekts von WFMS.

7. Informationen zu einer Organisationseinheit (Parameter: Organisationseinheit)

Bei dieser Auflistung ist zu beachten, dass abhängig vom jeweiligen WFMS bestimmte Objekte ggf. nicht vom WFMS unterstützt werden und somit bei der Konfiguration von Kontexten für ein konkretes WFMS die obige Liste angepasst werden muss. Ferner verzichten wir hier und auch für die weiteren Komplexitätsstufen jeweils auf eine Spezifikation der Attribute, die in den einzelnen Systemdokumenten angezeigt werden sollen, da auch diese nur für ein konkretes WFMS festgelegt werden können.

Komplexitätsstufe 2: Einfache Auswertungen von Laufzeitinformationen

Systemdokumente der Komplexitätsstufe 2 enthalten Informationen zu dynamischen Aspekten der Prozessdurchführung, die direkt aus den vom WFMS verwalteten Informationen abgeleitet werden können. Die Standardbibliothek soll dazu die folgenden Dokumenttypen enthalten:

1. Informationen zu einem Geschäftsfall: (Parameter: Geschäftsfall)

Mit Hilfe der historischen Information zu einem Geschäftsfall können die Bearbeiter bereits durchgeführter Aktivitäten festgestellt werden, und es können somit Ansprechpartner für Rückfragen identifiziert werden.

2. Informationen zu allen laufenden oder abgeschlossenen Geschäftsfällen eines Prozesses: (Parameter: Prozess)

Die Darstellung von Informationen zu laufenden und/oder abgeschlossenen Geschäftsfällen erleichtert die Suche eines Bearbeiters nach ähnlichen Geschäftsfällen, insbesondere wenn neben systemspezifischen Informationen, wie „Name“, „Erstellungszeitraum“ oder „Priorität“, auch prozessspezifische Variablen, wie im Fall des Prozesses „Contract Management“ z.B. „Kundenname“ und „Discount“, angezeigt werden.

3. Informationen zu allen Bearbeitern, denen aktuell in ihrer Arbeitsliste die Ausführung einer bestimmten Aktivität innerhalb eines bestimmten Geschäftsfall angeboten wird: (Parameter: Aktivität, Geschäftsfall)

Mit Hilfe dieser Informationen ist es für einen Bearbeiter möglich, die Menge der Personen festzustellen, die in einem Geschäftsfall eine bestimmte Aktivität ebenfalls zur Ausführung in ihrer Arbeitsliste haben, um somit Fragen der Arbeitsverteilung direkt mit den Betroffenen klären zu können.

4. Informationen zu allen aktiven Bearbeitern: (Parameter: keine)

Die Darstellung der gerade aktiven, d.h. beim WFMS angemeldeten Bearbeiter ist bei der Suche nach Ansprechpartnern im Fall dringender Rückfragen hilfreich.

Die bisher aufgeführten Informationen stellen heutige WFMS den Bearbeitern lediglich teilweise zur Verfügung. Die Informationen werden dabei den Bearbeitern oftmals über verschiedene Benutzerschnittstellen bzw. Client-Anwendungen angeboten, was die Nutzung erschwert (bspw. können in CSE Workflow bestimmte Informationen zum Organisationsaspekt

nicht über den Workflow-Client zur Bearbeitung von Prozessen, sondern lediglich über ein Werkzeug zur Organisationsmodellierung eingesehen werden). Diese Gründe rechtfertigen die Aufnahme dieser Informationen in die Standardbibliothek von Systemdokumenttypen.

Komplexitätsstufe 3: Statistische Auswertungen von Laufzeitinformationen

In diesen Dokumenten werden Informationen bereitgestellt, die sich auf dynamische Aspekte der Prozessdurchführung beziehen und mittels einfacher statistischer Berechnungen erzeugt werden. Hiermit soll dem Bearbeiter eine weitere Möglichkeit der Entscheidungsunterstützung durch das System angeboten werden. Folgende Dokumenttypen werden bereitgestellt:

1. Häufigkeit der Durchführung einer Aktivität durch die einzelnen Bearbeiter in Geschäftsfällen:

(Parameter: Aktivität)

Diese Information dient insbesondere zur Identifikation von Experten für einzelne Aktivitäten, indem Personen festgestellt werden, die eine bestimmte Aktivität bereits besonders häufig durchgeführt haben.

2. Informationen, welche Aktivitäten in welchen Geschäftsfällen als Folgeaktivität einer bestimmten Aktivität ausgeführt wurden:

(Parameter: Aktivität)

Erfolgt in einer Aktivität aufgrund des Prozessmodells eine Entscheidung über den weiteren Verlauf eines Geschäftsfalles, können zur Entscheidungsunterstützung Informationen hilfreich sein, die angeben, wie oft welche Alternative in früheren Geschäftsfällen ausgewählt wurde und insbesondere in welchen Geschäftsfällen dies jeweils der Fall war. Der Bearbeiter kann nun die Charakteristik konkreter Geschäftsfälle untersuchen, um seine Entscheidung zu treffen.

Die Standardbibliothek soll durch den Administrator erweitert werden können, indem dieser weitere Dokumenttypen erstellt und in die Bibliothek einfügt.

Bei Dokumenttypen der Komplexitätsstufe 2 und 3, die sich auf eine Menge von Geschäftsfällen beziehen, sollte in jedem zugehörigen Systemdokument für den Bearbeiter die Möglichkeit bestehen, die Menge der zu untersuchenden Geschäftsfälle geeignet einzuschränken. Dazu können die in Abschnitt 6.3.5 eingeführten Bedingungen prinzipiell genutzt werden, um die Menge der Geschäftsfälle, über die eine bestimmte Auswertung durchgeführt werden soll, einzuschränken. Es können sowohl systemspezifische als auch prozessspezifische Teilbedingungen formuliert werden, allerdings nur, soweit sie sich auf die Attribute bzw. Variablen von Geschäftsfällen (und nicht auf Aktivitätsausführungen) beziehen.

Ein Bearbeiter im Prozess „Contract Management“ benötigt z.B. weitere Informationen zur Berechnung einer für das Unternehmen optimalen Rabattierung in einem konkreten Geschäftsfall und sucht einen Experten zur Erörterung der möglichen Alternativen. Dazu öffnet er ein Systemdokument, das die Häufigkeit der Durchführung der Aktivität „Rabattanfrage erstellen“ durch die einzelnen Bearbeiter anzeigt. Um nun einen Experten zu finden, der eine

möglichst große Anzahl von Geschäftsfällen mit ähnlicher Charakteristik zum aktuellen Problem bereits bearbeitet hat, schränkt er in diesem Systemdokument die auszuwertende Menge der Geschäftsfälle mit der Bedingung „Priorität = 1 UND Discount \geq 10000“ ein.

Die Erstellung eines Systemdokuments integriert in die Kontextbearbeitung kann auf zwei Arten möglich sein: Wird in einem Kontext ein Systemdokument instanziiert, dessen Parameter bzgl. des Typs jeweils mit entsprechenden kontextbildenden Objekten des Kontexts übereinstimmen, wird das Systemdokument mit den Werten der entsprechenden kontextbildenden Objekte parametrisiert. Wird also ein Systemdokument, das Informationen zu einem bestimmten Geschäftsfall enthält (Parameter: Geschäftsfall), in einem Kontext erzeugt, der durch die kontextbildenden Objekte „Geschäftsfall: Fa. Siemens“ und „Aktivität: Rabattanfrage erstellen“ festgelegt ist (vgl. Abbildung 33), zeigt das Dokument automatisch die Historie und weitere Informationen zu diesem Geschäftsfall an. Können aber die Parameter des Dokuments durch den Kontext nicht abgedeckt werden, muss der Bearbeiter die fehlenden Werte manuell eingeben. Durch diese Möglichkeiten kann der Bearbeiter zur Laufzeit aus der Standardbibliothek Systemdokumente instanziiieren und in Kontexten speichern.

In Bezug auf die im vorherigen Abschnitt diskutierten Aspekte zur Beschreibung des Informationsmodells zu Kontextdokumenten sind noch einige Abhängigkeiten zu betrachten. Falls ein Kontextdokument über das Attribut „Typ“ als Systemdokument spezifiziert wird, sind dadurch weitere Attribute automatisch vorbelegt. Diese Abhängigkeiten werden im Folgenden aufgeführt:

- Identifikation: Wird vom System vergeben
- Exaktheit: = „Exakte Beschreibung“
- Verantwortlich: = „System“
- Autor: = „System“
- Gültigkeitszeitraum: Das Dokument ist immer gültig. Es kann kein Gültigkeitszeitraum vom Bearbeiter angegeben werden.
- Revisionsrhythmus: = „niemals“; das Dokument ist stets auf dem aktuellen Stand.

7 Methodik zur Unterstützung wissensintensiver Prozesse durch Workflow-Management-Systeme

Die zentrale Anforderung in Bezug auf die organisatorische Einbettung von Kontexten besteht in der geeigneten Integration sekundären Wissens und damit des Kontextkonzepts in den Prozesslebenszyklus. Abschnitt 5.5 gibt dazu einen Überblick über die einzelnen Aufgaben, die in Bezug auf das Management prozessbezogenen Wissens in den Phasen des Prozesslebenszyklus von Belang sind. In diesem Kapitel soll diese Fragestellung vertieft werden, indem konkret dargestellt wird, welche Aufgaben unterschiedliche Gruppen bzw. Rollen, wie z.B. Modellierer, Kontextverantwortliche oder Bearbeiter, in den einzelnen Phasen des Prozesslebenszyklus in Bezug auf die Verwaltung von Kontexten wahrnehmen. Dabei soll deutlich werden, mit welchen insbesondere organisatorischen Mitteln die kooperative Erstellung und Bewahrung von sekundärem Wissen sichergestellt werden kann.

Abschnitt 7.1 stellt dazu organisatorische Maßnahmen zur Verwaltung von Kontexten vor und zeigt die Integration dieser Maßnahmen in den Prozesslebenszyklus auf. Darauf aufbauend nimmt Abschnitt 7.2 das Beispiel „Contract Management“ (Abschnitt 4.2) auf und stellt dar, wie mit Hilfe von Kontexten eine verbesserte Prozessbearbeitung erreicht werden kann.

7.1 Integration von Maßnahmen zur Verwaltung von Kontexten in den Prozesslebenszyklus

Abschnitt 2.1.2 erläutert die einzelnen Phasen des Prozesslebenszyklus. Wir werden im Folgenden eine erweiterte Sicht auf diesen Prozesslebenszyklus vorstellen, die sowohl zusätzliche Phasen zur Verwaltung von Kontexten enthält (in der Abbildung grau hinterlegt) als auch die beteiligten Rollen vorstellt (Abbildung 36). Wie bereits in Abschnitt 5.5 diskutiert, ist die Phase der Erhebung hier nicht von Interesse, da das während der Erhebung in Form von Interviewprotokollen, schriftlichen Befragungen von Mitarbeitern, etc. gewonnene Wissen nur indirekt für die Durchführung von Prozessen von Belang ist. Ebenfalls in der Abbildung nicht dargestellt ist die Aktivität der Konfiguration eines um die Möglichkeit zur Verwaltung von Kontexten erweiterten WFMS. Wie im letzten Kapitel erläutert, erfordert eine informationstechnische Umsetzung von Kontexten als Erweiterung eines konkreten WFMS z.B. die Auswahl der in diesem WFMS konkret zur Verfügung stehenden kontextbildenden Objekttypen oder die Auswahl der prozessbezogenen Attribute, die zur Definition von Sichtbarkeitsbedingungen geeignet sind. Diese Aspekte werden im nächsten Kapitel ausgeführt.

Im Folgenden geben wir einen Überblick über die einzelnen Phasen des erweiterten Prozesslebenszyklus. Dabei setzen wir für die erste Phase der Prozessmodellierung und der Initialisierung von Kontexten die vorherige Auswahl, Installation und Konfiguration eines um das Management von Kontexten erweiterten WFMS voraus.

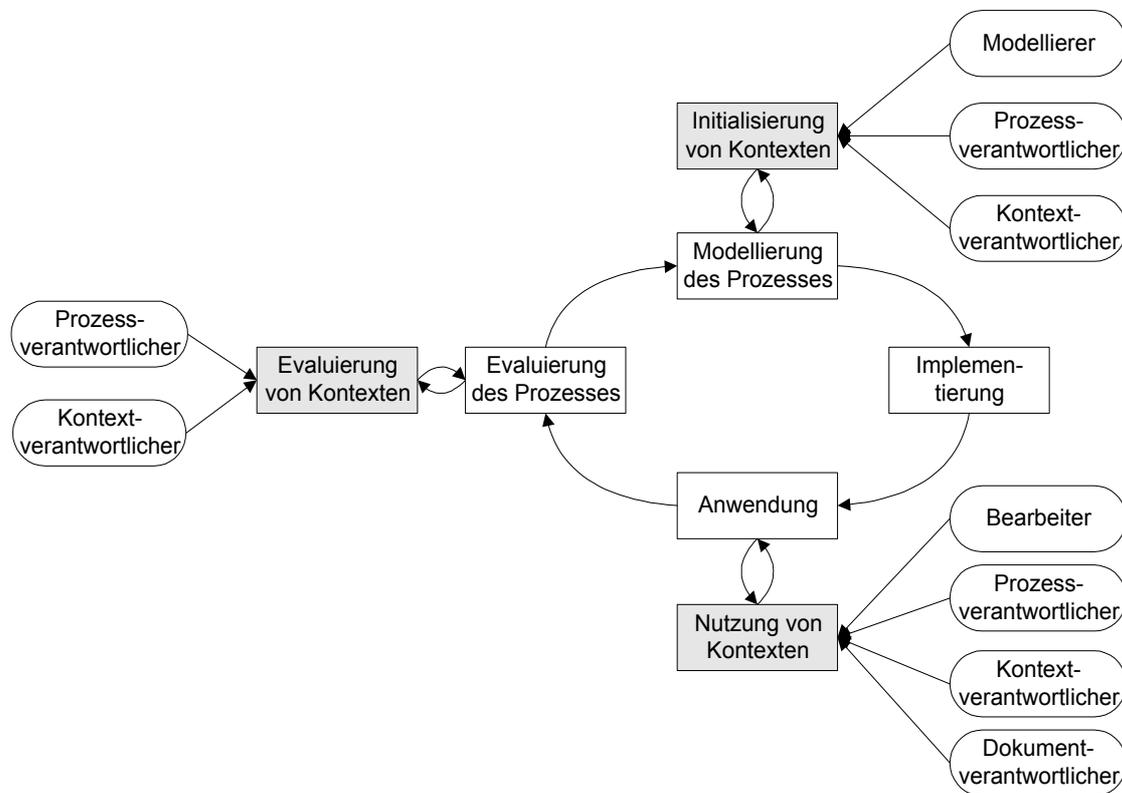


Abbildung 36: Verwaltung von Kontexten innerhalb des Prozesslebenszyklus

Parallel zur Modellierung des Prozesses im WFMS erfolgt die Initialisierung von Kontexten. Dies umfasst sowohl die automatische Erstellung der Kontexte aus dem Prozess- bzw. Organisationsmodell durch das System als auch manuelle Tätigkeiten zur Festlegung eines initialen Informationsbestands in Kontexten. Die Aufgabe des Modellierers liegt diesbezüglich u.a. darin, Dokumente, deren Relevanz für bestimmte Bearbeitungssituationen bereits zum Modellierungszeitpunkt bekannt ist, in die entsprechenden Kontexte einzufügen. Außerdem kann bereits zu diesem Zeitpunkt in Abstimmung mit den Beteiligten für bestimmte Kontexte ein Kontextverantwortlicher benannt werden, der eine besondere Verantwortung für den Aufbau und die Bewahrung von Wissen im entsprechenden Kontext besitzt. Eine detaillierte Beschreibung der einzelnen Maßnahmen erfolgt in Abschnitt 7.1.1.

Die Phase der Implementierung der Workflow-Management-Anwendung wurde lediglich aus Gründen der Vollständigkeit in den erweiterten Prozesslebenszyklus aufgenommen. Für diese Phase ergeben sich durch die Einführung von Kontexten keine weiteren Anforderungen.

Die Nutzung der Kontexte durch die Bearbeiter wird integriert in die Anwendung der Workflow-Management-Anwendung durchgeführt. Hier sind insbesondere Fragen der Wissensbewahrung relevant, die in Abschnitt 7.1.2 diskutiert werden.

Im Rahmen des kontinuierlichen Verbesserungsprozesses sind sowohl die Prozessbeschreibungen als auch die Kontextdokumente im Hinblick auf ihre Aufgabenangemessenheit zu evaluieren. Die spezifischen Methoden der Evaluation stehen in dieser Arbeit nicht im Zentrum der Betrachtung. Abschnitt 7.1.3 beschreibt daher lediglich das grundsätzliche Vorgehen bei der Evaluation von Kontexten.

7.1.1 Initialisierung von Kontexten

Zum Modellierungszeitpunkt wird sowohl das Prozess- und das Organisationsmodell festgelegt als auch die Erzeugung von Kontexten und das Einfügen von Kontextdokumenten vorgenommen. Dabei unterscheiden wir verschiedene Aktivitäten, die von unterschiedlichen Rollen durchgeführt werden (Abbildung 37). Diese werden im Folgenden erläutert.

Prozess- und Organisationsmodell festlegen

Zunächst wird vom Modellierer das Prozessmodell erstellt, und das Organisationsmodell wird um neue Personen, Rollen und Organisationseinheiten ergänzt, d.h. es wird das primäre Wissen im Prozess- bzw. Organisationsmodell repräsentiert. Die Durchführung dieser Aktivität beinhaltet keine kontextspezifischen Aspekte.

Kontexte aus Prozess- und Organisationsmodell erzeugen

Aus dem Prozessmodell und dem Organisationsmodell sowie der Menge der im konkreten WFMS zur Verfügung stehenden kontextbildenden Objekttypen erzeugt das WFMS die entsprechenden Kontexte, die zunächst leer sind, also keine Kontextdokumente enthalten (das Prinzip der Bildung von Kontexten ist in Abschnitt 6.2.2 erläutert).

Weitere Kontexte manuell erzeugen

Bestimmte Kontexttypen können nicht aus dem Prozess- bzw. Organisationsmodell abgeleitet werden. Da der kontextbildende Objekttyp „Aktivitätsblock“ im Allgemeinen durch die Prozessmodellierung von WFMS nicht abgebildet werden kann, muss der Modellierer über ein entsprechendes Werkzeug die Möglichkeit besitzen, Aktivitätsblöcke zu definieren. Das System legt dann die entsprechenden Kontexte automatisch an.

Eine Besonderheit ergibt sich bzgl. des Kontexttyps „Werkzeug“. Bei dokumentenorientierten WFMS lassen sich im Allgemeinen die Werkzeuge, mit denen Instanzen der definierten Dokumentvorlagen bearbeitet werden, direkt aus den Modellinformationen auslesen. Wie in Abschnitt 2.1.3 beschrieben, bieten WFMS, wie z.B. CSE Workflow, dem Benutzer die Möglichkeit, auch nicht im Modell spezifizierte Dokumente in der Umlaufmappe abzulegen. Dadurch ergibt sich der Fall, dass während der Prozessdurchführung zum Bearbeiten dieser Dokumente Werkzeuge, wie z.B. Office-Produkte, benutzt werden, ohne dass diese Werkzeuge im Prozessmodell festgelegt sind. Der Modellierer kann daher zusätzliche Kontexte zu Werkzeugen anlegen, um die Repräsentation sekundären Wissens zu diesen nicht im Prozessmodell enthaltenen Werkzeugen zu ermöglichen.

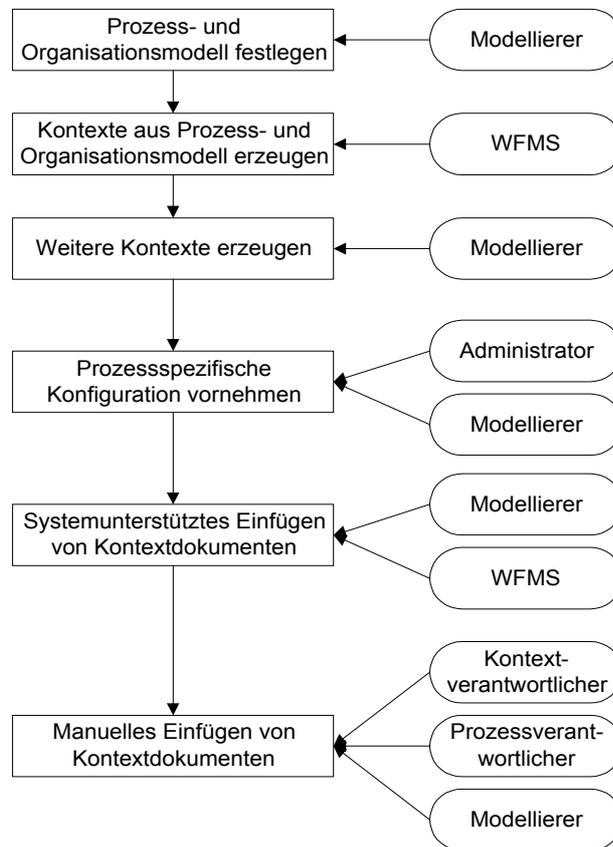


Abbildung 37: Aktivitäten bei der Initialisierung von Kontexten

Prozessspezifische Konfiguration vornehmen

Um prozessspezifische Sichtbarkeitsbedingungen zu Kontextdokumenten angeben zu können, ist vorher die Menge der diesbezüglich sinnvollen Prozessvariablen durch den Modellierer auszuwählen (Abschnitt 6.3.5). Zusätzlich sind die Variablen mit einem für den Benutzer verständlichen Namen zu versehen, da Prozessvariablen oftmals aufgrund von Systemeinschränkungen keine aussagekräftigen Namen besitzen (beispielsweise erlaubt das System COSA Workflow die Identifikation einer Prozessvariablen mit lediglich bis zu zwei Zeichen). Der Administrator des Systems stellt daher sicher, dass eine Verknüpfung zwischen dem logischen Variablennamen und der zugehörigen vom WFMS verwalteten Prozessvariablen angelegt wird.

Systemunterstütztes Einfügen von Kontextdokumenten

In diesem Schritt fügt das WFMS die Systemdokumente der Komplexitätsstufe 1 automatisch in die entsprechenden Kontexte ein. Dabei handelt es sich um

- Informationen zu einem Prozess,
- Informationen zu einer Aktivität,

- Informationen zu einer Dokumentvorlage,
- Informationen zu einem Werkzeug,
- Informationen zu einem Bearbeiter,
- Informationen zu einer Rolle,
- Informationen zu einer Organisationseinheit und
- Informationen zu einem Geschäftsfall.

Das System erstellt somit eine initiale Menge von Kontextdokumenten, indem in jedem Prozess-Kontext ein Systemdokument mit Informationen zu diesem Prozess angelegt wird, in jedem Aktivitäts-Kontext ein Systemdokument mit Informationen zu der Aktivität angelegt wird, etc. Über die Konfiguration der Standardbibliothek können allerdings bestimmte Systemdokumente ausgeschlossen werden, falls das WFMS keine aussagekräftigen Informationen zu dem entsprechenden kontextbildenden Objekt verwaltet. So besteht z.B. bei dem WFMS CSE Workflow die Menge der zu einer Aktivität verwalteten Informationen lediglich aus einer Kurzbezeichnung und einer Langbezeichnung. Diese Informationsmenge rechtfertigt somit nicht die Erstellung entsprechender Systemdokumente und deren Einbindung in die Menge der Aktivitäts-Kontexte.

Des Weiteren analysiert das WFMS die Struktur des definierten Prozesses und bietet dem Modellierer an, in bestimmten Kontexten Dokumente einzufügen. In folgenden Kontexten macht eine solche Bereitstellung von Dokumenten zum Modellierungszeitpunkt Sinn:

- Aktivitäts-Kontext, falls die zugehörige Aktivität eine Entscheidung beinhaltet:

Wird in einer Aktivität eine Entscheidung über den weiteren Verlauf des Geschäftsfalls getroffen (im Prozessmodell sind mehrere potentielle Folgeaktivitäten definiert), kann diese Aktivität vom System identifiziert werden. Das System kann nun dem Modellierer anbieten, die folgenden Dokumente im entsprechenden Aktivitäts-Kontext bereitzustellen:

Erstens sollte ein Systemdokument, das die Häufigkeit der Durchführung der Aktivität durch die einzelnen Bearbeiter angibt, im Kontext bereitstehen. Zweitens ist ein Systemdokument sinnvoll, das Informationen beinhaltet, welche Aktivitäten in welchen Geschäftsfällen als Folgeaktivität der aktuellen Aktivität durchgeführt wurden. Drittens ist schließlich bei komplexen Aktivitäten und zugehörigen Entscheidungen ein Dokument relevant, in das der Bearbeiter nach der Ausführung der Aktivität die Gründe für seine Entscheidung eintragen kann (die Spezifikation der aktiven Bereitstellung durch das System erfolgt ähnlich zu Beispiel 2 in Abschnitt 6.3.6 lediglich ohne Angabe einer Sichtbarkeitsbedingung).

Mit Hilfe der angegebenen Dokumente erhält der Bearbeiter einerseits Informationen über den Ablauf von Geschäftsfällen bzgl. der aktuellen Aktivität und über Experten, die er in einem komplexen Fall kontaktieren kann. Andererseits gibt ihm das System die Möglichkeit, seine Erfahrungen zu dokumentieren und als Entscheidungsunterstützung in späteren Geschäftsfällen bereitzustellen.

- Prozess-Kontext

Oftmals gehen bei der Bearbeitung von Prozessen gewonnenen Erfahrungen verloren, da in heutigen WFMS im Allgemeinen keine Möglichkeit besteht, diese Erfahrungen zu hinterlegen und für spätere Geschäftsfälle verfügbar zu machen. Mit Hilfe eines Dokuments, das aktiv vom System bei der Beendigung des Geschäftsfalls bereitgestellt wird (Abschnitt 6.3.6 und Tabelle 13) und dazu dient, Erfahrungen, die während der Bearbeitung gesammelt wurden, zu dokumentieren, kann dieser geforderte Wissenstransfer gefördert werden.

Die Entscheidung, ob die beschriebenen Dokumente jeweils in den entsprechenden Kontext eingefügt werden, sollte letztlich vom Modellierer getroffen werden. Das System kann zwar aufgrund des Prozessmodells Vorschläge bzgl. einzufügender Dokumente generieren, die letztendliche Entscheidung aufgrund z.B. der Komplexität einzelner Aktivitäten oder allgemein der Semantik des Prozesses kann jedoch nur der Modellierer treffen.

Manuelles Einfügen von Kontextdokumenten

Bestimmte sekundäre Wissensbestände sind bereits zum Modellierungszeitpunkt bekannt und können somit bereits vom Modellierer in die geeigneten Kontexte als Dokumente eingefügt werden. In Bezug auf die in Tabelle 6 angegebenen Standardtypen von Kontextdokumenten handelt es sich dabei neben den bereits angesprochenen Systemdokumenten um Ziele, Hilfedokumente, Vorgaben bzgl. der Bearbeitung, Bearbeitungsfristen oder allgemeine Hintergrundinformationen. Wichtig ist in diesem Zusammenhang die Rolle des Kontextverantwortlichen: Für Kontexte, die eine große Anzahl von Kontextdokumenten bzw. eine hohe Änderungshäufigkeit der Kontextdokumente erwarten lassen, sollte bereits zum Modellierungszeitpunkt ein Kontextverantwortlicher benannt werden, der zusätzlich zum Modellierer relevante Informationen in den entsprechenden Kontext einstellt. Beispiele für Kontexte, zu denen bereits zur Modellierung ein Kontextverantwortlicher benannt werden sollte, sind der Prozess-Kontext (hier bietet sich der Prozessverantwortliche als Kontextverantwortlicher an) oder Kontexte zu wissensintensiven Aktivitäten, die den eigentlichen „Kern“ des Prozesses bilden (hier sollte jeweils ein Mitarbeiter mit großer Erfahrung bzgl. der Bearbeitung dieser Aktivität als Kontextverantwortlicher bestimmt werden). Die Entscheidung, für welche Kontexte ein Kontextverantwortlicher bereits zum Modellierungszeitpunkt bestimmt wird, trifft der Modellierer in Absprache mit dem Prozessverantwortlichen.

7.1.2 Nutzung von Kontexten

Die Nutzung von sowohl primärem als auch sekundärem Wissen erfolgt integriert in die Prozessbearbeitung. In einem um Kontexte erweiterten WFMS muss daher die Möglichkeit bestehen, auf einfache Weise über den Workflow-Client auf Kontexte und das darin gespeicherte sekundäre Wissen zugreifen zu können. Dabei haben die Bearbeiter, wie in Kapitel 6 beschrieben, die Möglichkeit, vorhandene Kontextdokumente zu lesen, ihre Dokumentbeschreibung zu ändern, Kontextdokumente aus Kontexten zu löschen oder neue Dokumente in Kontexte einzufügen, falls die entsprechenden Berechtigungen vorhanden sind.

Beim Einfügen bzw. der Modifikation von Dokumenten in Kontexten können folgende Fälle unterschieden werden: Viele prozessbezogene Informationen sind bereits im Unternehmen in verschiedenen Informationssystemen vorhanden, ohne dass diese prozessbezogenen Informationen in Kontexten bereitstehen und damit als relevante Wissensquelle während der Prozessbearbeitung zur Verfügung stehen. Eine Hauptaufgabe beim kooperativen Wissensaufbau durch die Bearbeiter besteht daher darin, relevante vorhandene Dokumente in die geeigneten Kontexte einzufügen. Darüber hinaus ist es erforderlich, einen Teil des impliziten Wissens der Bearbeiter zu explizieren und in Form von Kontextdokumenten zu dokumentieren. Dies geschieht dadurch, dass Bearbeiter vorhandene Kontextdokumente modifizieren oder Wissen in Form von neuen Dokumenten in Kontexte einstellen, um z.B. Erfahrungen weiterzugeben. Schließlich besteht die Möglichkeit, aus der in Abschnitt 6.4 beschriebenen Standardbibliothek Systemdokumente zu instanziiieren und in Kontexten abzulegen, um relevante Informationen aus dem WFMS zur Entscheidungsunterstützung bereitzustellen.

Bei der Nutzung von Kontexten ist insbesondere die Wissensbewahrung, also das Sicherstellen der Relevanz und der Aktualität der Kontextdokumente, von Interesse. Diese muss durch organisatorische Maßnahmen und Verantwortlichkeiten gesichert werden. Dazu werden verschiedene organisatorische Rollen eingeführt, die in Abbildung 38 dargestellt werden:

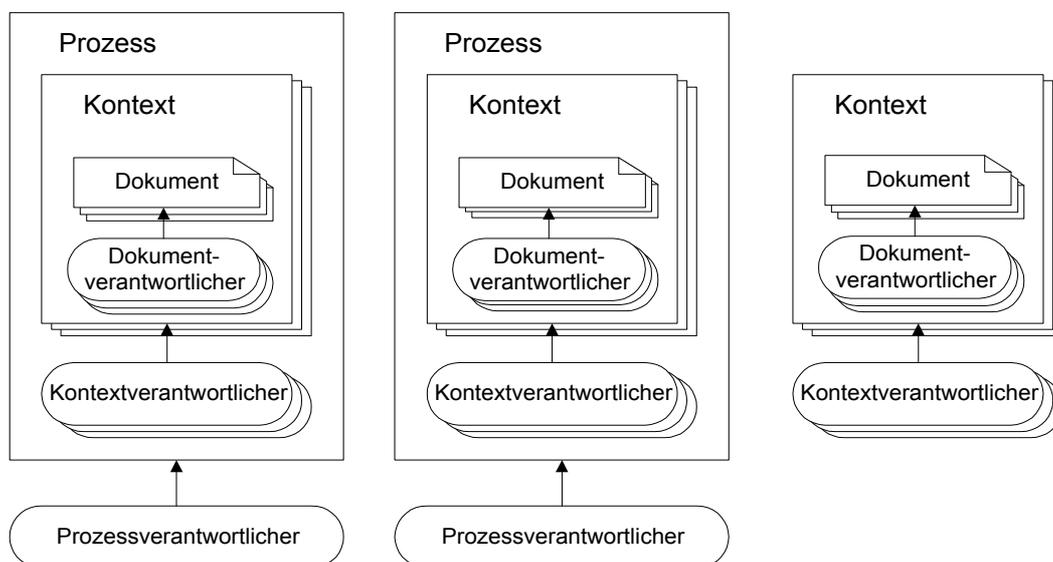


Abbildung 38: Hierarchie von Verantwortlichkeiten zur Wissensbewahrung in Kontexten

Wie in Abschnitt 6.3.3 (Aspekt „Lebenszyklus“ der Beschreibung von Kontextdokumenten) erläutert, bezeichnet das Attribut „Verantwortlicher“ die Person, die für die Aktualität eines Kontextdokuments verantwortlich ist (im Folgenden als „Dokumentverantwortlicher“ bezeichnet). Auch in Bezug auf einen gesamten Kontext kann ein Kontextverantwortlicher sowohl bereits zum Modellierungszeitpunkt (Abschnitt 7.1.1) als auch während der Prozessdurchführung bestimmt werden. Dieser fungiert als Wissensmanager für den Kontext und ist insbesondere dafür verantwortlich, dass der Wissenstransfer für den Kontext funktioniert, d.h. dass

relevante Dokumente bereitgestellt werden und diese auch genutzt werden. Schließlich ist der Prozessverantwortliche dafür zuständig, sowohl während der Modellierung als auch bei der Prozessdurchführung neues sekundäres Wissen, wie z.B. Ausnahmebehandlungen, Vorgaben zu Bearbeitungsfristen, Richtlinien für die Bearbeitung von Aktivitäten oder Entscheidungsspielräume, direkt in die entsprechenden Kontexte einzufügen bzw. die entsprechenden Kontextverantwortlichen zu benachrichtigen. Da nicht alle Kontexte inhaltlich einem übergeordneten Prozess zugeordnet werden können (dies ist z.B. der Fall bei Kontexten zu Objekten der Organisationsstruktur), existiert nicht zu allen Kontexten ein Prozessverantwortlicher. In diesem Fall sind die Aufgaben des Prozessverantwortlichen vom Kontextverantwortlichen zu übernehmen. Die einzelnen Rollen und ihre Verantwortlichkeiten für die Wissensbewahrung in Kontexten werden im Folgenden genauer erläutert.

Dokumentverantwortlicher

Bei dem Dokumentverantwortlichen handelt es sich standardmäßig um den Ersteller des Dokuments, also um die Person, die das Dokument in den entsprechenden Kontext eingefügt hat. Über das Attribut „Verantwortlicher“ kann aber auch eine andere Person die Rolle des Dokumentverantwortlichen übernehmen. Für jedes Kontextdokument muss ein Dokumentverantwortlicher existieren.

Die Aufgabe des Dokumentverantwortlichen besteht in der Prüfung sowohl der Aktualität des Dokumenteninhalts als auch der Relevanz des Dokuments für den Kontext, in dem es sich befindet. Hinweise auf die Relevanz eines Dokuments erhält der Dokumentverantwortliche durch die Attribute „Relevanz“ (von den Bearbeitern vorgenommene Bewertungen; Abschnitt 6.3.2) und Historie (statistische Daten über Häufigkeit des Zugriffs auf das Dokument, etc; Abschnitt 6.3.3). Ist die Aktualität oder die Relevanz nicht ausreichend, ist der Dokumentverantwortliche dafür zuständig, das Kontextdokument zu modifizieren bzw. die Modifikation anzustoßen, das Dokument aus dem Kontext zu löschen oder in einen anderen Kontext zu übertragen. Außerdem kann der Dokumentverantwortliche über eine Modifikation der Attribute zu einem Kontextdokument auf sich ändernde Inhalte oder Funktionen des Dokuments reagieren, indem er z.B. Sichtbarkeitsbedingungen für ein Dokument setzt (Abschnitt 6.3.5) oder eine aktive Bereitstellung des Dokuments (Abschnitt 6.3.6) festlegt. Die Durchführung dieser Maßnahmen ist nicht auf den Dokumentverantwortlichen beschränkt (falls dies nicht durch explizite Angaben im Aspekt „Adressaten und Berechtigungen“ festgelegt ist), da der beschriebene Ansatz eine möglichst breite Beteiligung der Bearbeiter an sowohl Wissensaufbau als auch Wissensbewahrung ermöglichen soll. Trotzdem ist eine ausgezeichnete Rolle und Verantwortung für Dokumente sinnvoll, da die Erfahrung aus Wissensmanagementprojekten zeigt, dass die Wissensbewahrung ohne explizite organisatorische Regelungen zumeist zum Scheitern verurteilt ist.

Eine Unterstützung der Aufgaben des Dokumentverantwortlichen bieten die beiden Attribute „Gültigkeitszeitraum“ und „Revisionsrhythmus“ (Abschnitt 6.3.3). Ist das Enddatum des Gültigkeitszeitraums eines Kontextdokuments überschritten oder das im Revisionsrhythmus spezifizierte Zeitintervall abgelaufen, wird der Dokumentverantwortliche automatisch vom System informiert. Möglich ist z.B. eine Benachrichtigung per E-Mail, die zu jedem der betroffenen Dokumente sowohl den Dokumententitel und den Grund der Information als auch einen direkten Zugriff auf den zugehörigen Kontext per Hyperlink beinhaltet. Der Dokumentverant-

wortliche kann somit direkt auf die Dokumente in den zugehörigen Kontexten zugreifen und die notwendigen Maßnahmen zur Wissensbewahrung ergreifen, ohne dass sich ein Suchaufwand ergibt. Dadurch dass der kleinste spezifizierbare Revisionsrhythmus „täglich“ beträgt, ist gewährleistet, dass jeder Dokumentverantwortliche höchstens einmal am Tag vom System notifiziert wird (im realen Betrieb wird dies im Allgemeinen sogar wesentlich seltener der Fall sein, da der überwiegende Anteil der Dokumente einen Revisionsrhythmus und Gültigkeitszeitraum von mindestens einigen Wochen bis Monaten haben wird). Der Aufwand zur Wissensbewahrung für jeden einzelnen Dokumentverantwortlichen kann daher als gering angenommen werden.

Kontextverantwortlicher

Wie bereits in Abschnitt 6.2.2 ausgeführt, ist die Rolle des Kontextverantwortlichen nur für Kontexte vorzusehen, die initial zum Modellierungszeitpunkt als zentral für die Prozessdurchführung angesehen werden oder die sich zur Laufzeit durch eine hohen Wissenstransfer auszeichnen. Hinweise für einen hohen Wissenstransfer sind beispielsweise eine große Anzahl von Dokumenten, die sich durchschnittlich in einem Kontext befindet, und eine hohe Änderungshäufigkeit von Dokumenten.

Der Kontextverantwortliche kann bereits zum Modellierungszeitpunkt durch den Modellierer und den Prozessverantwortlichen bestimmt werden (Abschnitt 7.1.1). Für Kontexte, die unabhängig von konkreten Prozessen sind und an kontextbildende Objekte der Organisationsstruktur gebunden sind, sind die Kontextverantwortlichen bereits implizit vorgegeben, oder sie können jeweils aus der Menge der Beteiligten bestimmt werden: Bei einem Kontext, der Wissen über einen bestimmten Bearbeiter enthält, sollte dieser auch der Kontextverantwortliche sein. Handelt es sich um einen Kontext, der Wissen über eine Rolle oder Organisationseinheit enthält, sollte ein Mitarbeiter mit der entsprechenden Rolle bzw. ein Mitarbeiter der Organisationseinheit die Aufgabe des Kontextverantwortlichen übernehmen. Bei Kontexten, die sich auf einen konkreten Prozess bzw. dessen Geschäftsfälle, Aktivitäten, etc. beziehen, sollte der Prozessverantwortliche zur Laufzeit die Kontexte identifizieren, für die die Rolle des Kontextverantwortlichen explizit besetzt werden muss.

Die Aufgaben des Kontextverantwortlichen im Rahmen des Wissensaufbaus und der Wissensbewahrung sind die Folgenden: Der Kontextverantwortliche dient als Wissensmanager und allgemeiner Ansprechpartner für die Nutzer des Kontexts, falls z.B. Bearbeiter unsicher sind, ob ein bestimmtes Dokument für ihre Bearbeitungssituation prinzipiell relevant ist oder ein neues Dokument aufgrund seines Inhalts und seiner Funktion in den Kontext eingefügt werden sollte. Darüber hinaus überwacht der Kontextverantwortliche, ob zu abgelegten Wissensbedarfen (vgl. Attribut „Exaktheit“ in Abschnitt 6.3.3) entsprechende Dokumente bereitgestellt werden, d.h. ob die Wissensbedarfe der Nutzer erfüllt werden. Ist dies nicht der Fall und handelt es sich um einen nicht erfüllten Wissensbedarf hoher Relevanz, sollte er die Suche nach zugehörigem Wissen und dessen Einstellung als Kontextdokument anstoßen oder ggf. sogar selbst durchführen. Schließlich ist der Kontextverantwortliche dafür zuständig, bestimmte Informationen, die für seinen Kontext relevant sind und die sich auf Vorgaben der Bearbeitung, Bearbeitungsfristen oder Ziele der Bearbeitung beziehen, in den Kontext einzustellen und aktuell zu halten.

Prozessverantwortlicher

Die kontextbezogenen Aufgaben des Prozessverantwortlichen beziehen sich hauptsächlich auf die Phasen der Initiierung und Evaluation von Kontexten. Während der Nutzung ist der Prozessverantwortliche darüber hinaus für zwei Teilaufgaben zuständig: Sollten sich bestimmte Vorgaben, Fristen oder Ziele für den zugehörigen Prozess ändern, sind die entsprechenden Informationen jeweils in einem geeigneten Kontext ggf. unter Einbeziehung des Kontextverantwortlichen einzufügen bzw. vorhandene Informationen zu modifizieren. Außerdem muss der Prozessverantwortliche einen Überblick über die Nutzung der Kontexte erhalten, um z.B. die Ernennung neuer Kontextverantwortlicher anzustoßen. Zu diesem Zweck muss ihm das System historische Daten über relevante Aspekte, wie z.B. die durchschnittliche Anzahl von Dokumenten, die durchschnittliche Änderungshäufigkeit von Dokumenten oder die Anzahl der Zugriffe auf Dokumente, jeweils für die einzelnen Kontexte des Prozesses als Entscheidungsgrundlage liefern.

7.1.3 Evaluierung von Kontexten

Die Evaluierung von Kontexten wird integriert in die Prozessevaluierung und damit die kontinuierliche Prozessverbesserung durchgeführt. Dabei sind zwei Aspekte zu unterscheiden, die die Evaluation und anschließende Prozessmodellierung bzw. Initiierung von Kontexten betreffen:

- Modifikationen des Prozessmodells oder des Organisationsmodells und die Auswirkungen dieser Modifikationen auf Kontexte
- Evaluation der Kontexte und der darin enthaltenen Kontextdokumente

Die Evaluation eines Prozesses führt oftmals zur Identifikation von Schwächen der Prozess- oder der Organisationsstruktur, die zu einer erneuten Modellierung der entsprechenden Aspekte führen. In diesem Zusammenhang kann es insbesondere zu einer Veränderung der kontextbildenden Objekte kommen, wenn z.B. eine neue Aktivität oder eine neue Dokumentvorlage in ein Prozessmodell eingefügt wird oder im Extremfall die Prozessdefinition komplett geändert wird. Es ist die Aufgabe des Prozessverantwortlichen und der entsprechenden Kontextverantwortlichen die bestehenden Kontextdokumente in die neue Kontextstruktur zu überführen. Wird z.B. eine Aktivität aus dem Prozessmodell entfernt, müssen lediglich die Kontexte, die an das zugehörige kontextbildende Objekt gebunden sind, auf zukünftig relevante Dokumente untersucht werden. Diese Dokumente müssen dann in anderen Kontexten abgelegt werden. Wird allerdings die Prozessstruktur in grundsätzlicher Weise geändert, ergeben sich weitere Aufwände. Eine automatische Überführung der bisherigen Kontextdokumente in die sich neu ergebende Kontextstruktur ist nicht möglich, da dem System nicht die volle Semantik sowohl der Kontextdokumente als auch der kontextbildenden Objekte bekannt sein kann.

Integriert in eine Evaluation der Prozess- oder Organisationsstruktur erfolgt eine Bewertung der Kontexte und der darin enthaltenen Kontextdokumente. Als Basis der Evaluation dienen einerseits die während der Nutzung der Dokumente von den Bearbeitern abgegebenen Bewertungen der inhaltlichen Relevanz eines jeden Dokuments. Andererseits sind die vom System gesammelten historischen Daten, wie z.B. die Änderungshäufigkeit eines Dokuments oder die Anzahl der Zugriffe auf ein Dokument, ein wichtiger Hinweis für die Relevanz. Wir wollen in

dieser Arbeit keine Sammlung von Evaluationskriterien erstellen, sondern die grundsätzlichen Möglichkeiten zur Modifikation von Kontextdokumenten im Rahmen der Evaluation aufführen:

- Das Kontextdokument wird aus dem Kontext entfernt und in das Prozessmodell aufgenommen.

Zeigt sich bei der Evaluation, dass die Bearbeiter die Relevanz des Dokuments als sehr hoch einschätzen und dass das Dokument sehr häufig zugegriffen wird, kann es sinnvoll sein, das Dokument aus dem Kontext zu entfernen und stattdessen an geeigneter Stelle in das Prozessmodell einzufügen. Aus sekundärem Wissen (Kontextdokument) wird also primäres Wissen (Prozessmodelldokument).

- Das Kontextdokument verbleibt im Kontext.

Liegt eine ausreichende Nutzung des Dokuments vor und wird das Dokument als relevant für die entsprechende Bearbeitungssituation angesehen, verbleibt es im Kontext. Darüber hinaus kann es sinnvoll sein, bestimmte Attribute des Dokuments zu ändern (also z.B. eine aktive Bereitstellung zu spezifizieren oder Sichtbarkeitsbedingungen zu ändern).

- Das Kontextdokument wird aus dem Kontext entfernt.

Ist die Nutzung und die Relevanz des Dokuments unzureichend, wird das Dokument aus dem Kontext entfernt. Falls das Dokument für eine andere Bearbeitungssituation relevant ist, wird es in den entsprechenden Kontext übertragen.

Die aufgeführten Maßnahmen werden von den Prozessverantwortlichen in enger Abstimmung mit den Kontextverantwortlichen durchgeführt.

7.2 Unterstützung des wissensintensiven Prozesses "Contract Management" durch Kontexte

In diesem Abschnitt nehmen wir das in Abschnitt 4.2 dargestellte Beispiel eines wissensintensiven Prozesses wieder auf und zeigen, wie mit Hilfe der in Kapitel 6 eingeführten Kontexte und der im vorherigen Abschnitt diskutierten organisatorischen Methodik eine im Vergleich zu heutigen WFMS verbesserte Unterstützung dieses Prozesses bereitgestellt werden kann. Dazu gehen wir anhand der in Abbildung 36 und Abbildung 37 dargestellten Phasen vor und zeigen exemplarisch auf, wie die Kernanforderungen des Beispiels „Contract Management“ umgesetzt werden können (die hierzu relevanten Abschnitte der Beschreibung des Beispiels werden an den entsprechenden Stellen aufgeführt).

Initialisierung von Kontexten

Zunächst wird vom Modellierer das Prozessmodell, wie in Abbildung 24 dargestellt, festgelegt. Aus diesem Prozessmodell und dem zugehörigen Organisationsmodell werden daraufhin die entsprechenden Kontexte erzeugt. Im nächsten Schritt legt der Modellierer manuell weitere Kontexte an, die nicht aus den Modellen abgeleitet werden können:

Aspekt 1:

„Da der Kunde im Lauf des Gesprächs geäußert hatte, auch mit weiteren Wettbewerbern zu verhandeln, ist es wichtig, das im Kundengespräch gewonnene Wissen über Angebote der Wettbewerber auch weiteren Personen im Unternehmen verfügbar zu machen. [...] Außerdem kann dieses Wissen ebenfalls für die Entscheidung über eine Rabattanfrage relevant sein.“

Die Beschreibung liefert einen Hinweis darauf, dass bestimmtes Wissen für alle drei Aktivitäten „Rabattierung mit Kunden verhandeln“, „Rabattanfrage erstellen“ und „Rabattanfrage entscheiden“ relevant ist und zur Verfügung stehen sollte. Daher legt der Modellierer einen Aktivitätsblockkontext an, der die drei genannten Aktivitäten umfasst.

Aspekt 2:

„Bei der Erstellung einer Rabattanfrage kann es sich daher um eine komplexe Aufgabe handeln, bei der der Verkäufer - je nach einerseits Rabattierungsart, Sendungsaufkommen des Kunden, etc. und andererseits persönlicher Erfahrung - auf verschiedene Dokumente zurückgreifen muss.“

Um Sichtbarkeitsbedingungen für Dokumente in Abhängigkeit von Prozessvariablen, wie „Rabattierungsart“ oder „Sendungsaufkommen“, festlegen zu können, ist eine prozessspezifische Auswahl der diesbezüglich sinnvollen Prozessvariablen notwendig. Modellierer und Administrator stellen daher sicher, dass diese Prozessvariablen für die Definition von Sichtbarkeitsbedingungen zur Verfügung stehen.

Aspekt 3:

„Der Mitarbeiter im Preiskomitee erhält die Rabattanfrage und ist sich in diesem komplexen Fall unsicher, ob die Begründung des Verkäufers schlüssig ist. Für seine Entscheidungsfindung ist einerseits Wissen zu abgelaufenen Geschäftsfällen relevant, das von seiner Einschätzung der Rabattierung und des Verkäufers abhängt und damit nicht im Voraus planbar ist: Wurde zu diesem Kunden bereits vorher eine Vertragserstellung genehmigt oder abgelehnt? Welche Rabattanfragen hat der genannte Verkäufer in letzter Zeit eingereicht, und konnten diese genehmigt werden?“

Das System stellt in einem nächsten Schritt, wie in Abschnitt 7.1.1 geschildert, in Interaktion mit dem Modellierer bestimmte Dokumente initial in die Kontexte ein. Im gegebenen Beispiel ist insbesondere für die Aktivität „Rabattanfrage entscheiden“ eine Unterstützung durch Systemdokumente hilfreich. Daher werden in den entsprechenden Aktivitäts-Kontext Systemdokumente eingefügt, die die angegebenen Informationen über die Historie von Geschäftsfällen liefern.

Im letzten Schritt der Initialisierung von Kontexten werden Kontextverantwortliche benannt und bereits bekanntes Wissen in Form von Dokumenten in Kontexte eingefügt. Im gegebenen Beispiel werden für die wissensintensiven Aktivitäten „Rabattierung mit Kunden verhandeln“, „Rabattanfrage erstellen“ und „Rabattanfrage entscheiden“ bereits zu diesem Zeitpunkt Kontextverantwortliche eingesetzt. Die in Aspekt 2 genannten Dokumente, die Vorgaben, Entscheidungsspielräume oder Bearbeitungsfristen bzgl. der Erstellung einer Rabattanfrage beinhalten, werden nun jeweils in den Kontext der zugehörigen Aktivität eingefügt. Dabei werden insbesondere die in Abschnitt 6.3 beschriebenen Aspekte „Sichtbarkeit“ und „Bereitstellung“ genutzt, um einerseits Dokumente z.B. über Entscheidungsspielräume nur anzuzeigen, wenn der Kunde ein bestimmtes Sendungsaufkommen besitzt, bzw. um Dokumente aktiv bereits vor der Bearbeitung der Aktivität bereitzustellen und somit z.B. auf Fristen oder Ausnahmebehandlungen hinzuweisen.

Ein weiterer Aspekt betrifft die Weitergabe von Erfahrungen:

Aspekt 4:

Abschließend kontaktiert er einen erfahrenen Kollegen, der ihm eine Begründung für die Genehmigung der Rabat-
tanfrage liefert, die nicht in den offiziellen Dokumenten enthalten ist. Um die Gründe für seine Entscheidung für
ihn und seine Vorgesetzten nachvollziehbar zu machen, müssen diese expliziert werden. Außerdem ist es wichtig
nachzuhalten, wer in diesem Geschäftsfall als Experte fungiert hat. Auch dieses Wissen kann für die Bearbeitung
späterer Geschäftsfälle relevant sein.

Zur Umsetzung dieser Anforderung fügt der Modellierer ein leeres Dokument mit dem Titel
„Gründe für die Entscheidung über Rabattanfragen“ in den Kontext ein, der durch die Aktivi-
tät „Rabattanfrage entscheiden“ festgelegt ist. Über die Spezifikation der aktiven Bereitstel-
lung des Dokuments beim Eintreten des Ereignisses „Aktivität beendet“ wird der Mitarbeiter
des Preiskomitees durch das System aufgefordert (aber nicht gezwungen), seine Entscheidung
zu begründen. Über die Einschränkung der Adressatengruppe des Dokuments auf Mitglieder
der Organisationseinheit „Preiskomitee“ ist sichergestellt, dass lediglich berechnigte Mitarbei-
ter die getroffenen Entscheidungen einsehen können. Darüber hinaus werden weitere Doku-
mente, z.B. zu Werkzeugen (FAQs, Hilfesysteme, Probleme und Lösungen) oder zum
gesamten Prozess (Ziele, Fristen, Experten, etc.), in die entsprechenden Kontexte eingefügt.

Nutzung von Kontexten

Die Nutzung von Kontexten zum kooperativen Aufbau und zur kooperativen Bewahrung von
Wissen im Beispiel „Contract Management“ erfolgt gemäß den in Abschnitt 7.1.2 aufgeführ-
ten Möglichkeiten und Verantwortlichkeiten. Insbesondere der Zugriff auf dynamisch zur Lauf-
zeit entstehende Kontexte, wie z.B. Kontexte zu Geschäftsfällen, ist erst in dieser Phase
möglich. Im Folgenden sollen zwei im Beispiel genannte Anforderungen besonders betrachtet
werden.

Aspekt 5:

Der Verkäufer nimmt den Kontakt zum Kunden auf und vereinbart einen Termin, um über mögliche Rabattierungs-
alternativen zu verhandeln. Inzwischen hat der Marketing Manager eine neue Version des Dokuments zu mögli-
chen Rabattierungen bei Großkunden erarbeitet. Da die Änderung dieser Richtlinie dem Verkäufer ggf. neue
Spielräume bei der Verhandlung mit dem Kunden eröffnet (oder diese auch beschränkt), soll dem Verkäufer das
Dokument zum richtigen Zeitpunkt, also bei der Vorbereitung auf den Termin zugeleitet werden

Eine Umsetzung dieser Anforderung kann auf folgende Weise erfolgen: Das Dokument über
mögliche Rabattierungsalternativen wird nach der Änderung in den Kontext eingefügt, der
durch die Aktivität „Rabattierung mit Kunden verhandeln“ definiert ist. Es wird festgelegt,
dass das Dokument aktiv bereitgestellt werden soll, falls das Ereignis „Aktivität gestartet“ auf-
tritt. Die Wiederholbarkeit wird auf „einmalig“ festgelegt um sicherzustellen, dass jeder Bear-
beiter nur ein einziges Mal über die neue Richtlinie informiert wird.

Aspekt 6:

Das Informationssystem zur Erstellung von Rabattanfragen ist in einigen Bereichen kompliziert zu bedienen. Daher
bestehen bei vielen Mitarbeitern Wissensbedarfe bzgl. der Bedienung des Systems, der richtigen Reaktion auf vom
System angezeigte Meldungen, etc. Diese Wissensbedarfe sollten expliziert werden können, um einen entsprechen-
den zielgerichteten Wissensaufbau zu voranzutreiben.

Über das Attribut „Exaktheit“ der Dokumentbeschreibung können Wissensbedarfe explizit ausgezeichnet werden. Somit ist sowohl den Bearbeitern als auch dem entsprechenden Kontextverantwortlichen bekannt, zu welchen Themen Wissen kooperativ aufgebaut werden muss.

Auf die Evaluierung von Kontexten wird im Beispiel nicht gesondert eingegangen. Hier sind die im Abschnitt 7.1.3 skizzierten Möglichkeiten anwendbar.

D Informationstechnische Umsetzung und Realisierung eines erweiterten Workflow-Management-Systems zur Unterstützung wissensintensiver Prozesse

- 8 Erweiterung von Workflow-Management-Systemen zur Unterstützung von Kontexten
- 9 Realisierung

8 Erweiterung von Workflow-Management-Systemen zur Unterstützung von Kontexten

Um das beschriebene Konzept der Kontexte informationstechnisch umsetzen zu können, ist es notwendig, eine Erweiterung eines WFMS durch Integration eines prozessbezogenen OMIS vorzunehmen. In diesem Kapitel stellen wir ein entsprechendes Konzept vor, das aufbauend auf einer Anforderungsanalyse (Abschnitt 8.1) die Systemarchitektur erweiterter WFMS vorstellt und die Aufgaben der einzelnen Komponenten erläutert (Abschnitt 8.2). Wir werden im folgenden Kapitel dieses Konzept aufnehmen und anhand zweier Prototypen die Umsetzung des Architekturvorschlags basierend auf einem konkreten WFMS aufzeigen.

8.1 Anforderungsanalyse

Zunächst wird im folgenden Abschnitt geklärt, welche prinzipiellen Voraussetzungen WFMS erfüllen müssen, damit sie gemäß dem hier beschriebenen Ansatz um die Möglichkeit zur Verwaltung von Kontexten erweitert werden können. Anschließend wird eine anwendungsfallbezogene Analyse der Anforderungen an eine entsprechende Erweiterung von WFMS durchgeführt.

8.1.1 Anforderungen an WFMS

Ein Hauptziel des informationstechnischen Konzepts zur Erweiterung von WFMS um Kontexte besteht darin, dass das Konzept für eine möglichst große Klasse von WFMS anwendbar sein muss und somit nicht auf ein spezifisches Workflow-Management-Produkt ausgerichtet ist. Dennoch müssen einige wenige Annahmen über ein zu erweiterndes WFMS getroffen werden. Diese werden im Folgenden aufgeführt:

- Es muss sich um ein Client-/Server-basiertes System handeln:

Ein WFMS besteht gemäß dem Client-Server-Ansatz aus ggf. unterschiedlichen Workflow-Client-Komponenten, die mit einer Server-Komponente kommunizieren (analog z.B. zum Referenzmodell der WfMC, in der die Server-Komponente durch den Workflow Enaction Service repräsentiert wird). Die überwiegende Anteil existierender WFMS folgt grundsätzlich diesem Architekturmodell. Auch bei WFMS, die, wie z.B. WorkParty¹ von Siemens Nixdorf, keine Workflow-Server-Komponente beinhalten und bei denen der Workflow-Client direkt auf eine zentrale Datenbank zugreift, ist die im Folgenden beschriebene informationstechnische Umsetzung von Kontexten mit geringen Änderungen anwendbar.

- Der Workflow-Client muss erweiterbar sein:

Der Workflow-Client muss in geringem Umfang modifiziert werden (eine genauere Beschreibung erfolgt in Abschnitt 8.2.3). Dagegen kann der Workflow-Server unverändert in die neue Architektur übernommen werden. Der Workflow-Server muss lediglich bestimmte Funktionalitäten an seiner Schnittstelle bereitstellen (Abschnitt 8.2.7).

1. Die Entwicklung und der Vertrieb von WorkParty wurde mittlerweile eingestellt.

- Das Workflow-Metamodell sollte in wesentlichen Punkten den kontextbildenden Objekten entsprechen:

In Abbildung 32 wurde ein Vorschlag für eine Menge kontextbildender Objekttypen vorgestellt und darauf hingewiesen, dass die meisten WFMS nicht alle aufgeführten kontextbildenden Objekttypen unterstützen (z.B. unterstützen WFMS oftmals nur ein im Vergleich vereinfachtes Organisationsmodell). Um aber eine sinnvolle Menge von Kontexten bilden zu können, sollte das WFMS zumindest einen hohen Anteil der kontextbildenden Objekttypen beinhalten.

8.1.2 Anwendungsfallanalyse

In der Anwendungsfallanalyse werden die Interaktionen zwischen den Anwendern¹ und dem zu entwickelnden System beschrieben. Dabei beschreibt jeder Anwendungsfall die Durchführung eines in sich geschlossenen Arbeitsvorgangs [Oest98], ohne dass hier bereits auf einzelne Komponenten des Systems eingegangen wird. Die im Folgenden aufgeführten Anwendungsfälle nehmen also die in den beiden vorherigen Kapiteln genannten Interaktionsmöglichkeiten zwischen Anwender und System auf und fassen somit die funktionalen Anforderungen an eine Erweiterung von WFMS zusammen.

Erzeugen von Kontexten:

Aus der Menge der im WFMS zur Verfügung stehenden kontextbildenden Objekte erzeugt das WFMS sowohl zum Modellierungszeitpunkt (Abschnitt 7.1.1) als auch dynamisch zur Laufzeit (beim Start von Geschäftsfällen) die entsprechenden Kontexte ohne weitere Interaktion mit dem Anwender. Darüber hinaus besitzt der Anwender die Möglichkeit, manuell Kontexte zu Aktivitätsblöcken und Werkzeugen zu erzeugen.

Visualisierung von bzw. Navigation auf Kontexten:

Die Visualisierung der Kontexte erfolgt gemäß der in Abschnitt 6.2 vorgestellten Kontextstruktur. Dabei sind zwei Fälle zu unterscheiden (eine detaillierte Beschreibung gibt Abschnitt 8.2.2): Benötigt der Anwender Informationen, die zur Bearbeitung einer konkreten Aktivität eines Geschäftsfalls relevant sind, wird ihm der „aktuelle Kontext“ angezeigt. Dieser fasst bestimmte Kontexte zusammen, die für seine aktuelle Bearbeitungssituation von besonderer Relevanz sind.

Darüber hinaus ist es dem Anwender möglich, auf der Menge der Kontexte frei zu navigieren. Dafür stellt das System bestimmte Startpunkte und eine geeignete Vernetzung der Kontexte auf der Benutzeroberfläche bereit. In beiden Fällen gilt: Wählt der Bearbeiter einen Kontext aus, werden die entsprechenden Kontextdokumente angezeigt.

1. Wir unterscheiden in diesem Abschnitt nicht zwischen den einzelnen in Kapitel 7 aufgeführten Benutzergruppen und verwenden daher den allgemeinen Begriff „Anwender“.

Anzeigen und Modifizieren von Kontextdokumenten:

Das Anzeigen von Kontextdokumenten kann auf Basis einer pull- oder einer push-Aktivierung durchgeführt werden (Abschnitt 6.3.6): Wählt im Fall einer pull-Aktivierung der Anwender einen Kontext aus und besitzt er die entsprechende Berechtigung auf dem ausgewählten Kontext (Abschnitt 6.3.4), wird eine Liste der Kontextdokumente angezeigt, für die jeweils die Sichtbarkeitsbedingung (Abschnitt 6.3.5) erfüllt ist.

Falls der Anwender im Workflow-Client ein Ereignis auslöst, das eine aktive Bereitstellung einer Menge von Kontextdokumenten (push-Aktivierung) aus ggf. unterschiedlichen Kontexten bewirkt, wird auch in diesem Fall für jedes dieser Kontextdokumente die Berechtigung des Bearbeiters und die Sichtbarkeitsbedingung geprüft und eine Liste der sich qualifizierenden Kontextdokumente in einem separaten Fenster angezeigt. In beiden Fällen kann der Anwender jeweils ein Kontextdokument in der Liste auswählen, öffnen und modifizieren.

Manuelles Einfügen eines Dokuments in einen Kontext:

Beim Einfügen eines Dokuments in einen Kontext ist zu unterscheiden, ob das Dokument bereits in einem Kontext enthalten ist oder sich noch nicht in einem Kontext befindet: Im ersten Fall wählt der Anwender zunächst den Kontext aus, in den das Dokument eingefügt werden soll und spezifiziert daraufhin die Werte sowohl der kontextbezogenen Attribute als auch der dokumentbezogenen Attribute für dieses Dokument.

Im zweiten Fall wählt der Anwender zunächst den Kontext und das Kontextdokument aus, das in einen weiteren Kontext eingefügt werden soll. Draufhin gibt er den Zielkontext an und spezifiziert die Werte der kontextbezogenen Attribute für das Kontextdokument.

Eine Besonderheit ergibt sich bei Geschäftsfall-Kontexten: Ist die Bearbeitung eines Geschäftsfalls abgeschlossen, kann in einen entsprechenden Kontext kein Dokument mehr eingefügt werden. Der lesende Zugriff auf den Kontext ist allerdings weiterhin möglich.

Manuelles Einfügen eines Systemdokuments in einen Kontext:

Soll ein Systemdokument in einen ausgewählten Kontext eingefügt werden, zeigt das System zunächst die zur Verfügung stehenden Dokumenttypen der Standardbibliothek an. Der Anwender wählt einen Dokumenttyp aus, der daraufhin vom System instanziiert wird. Werden Parameter zur Spezifikation des Systemdokuments benötigt, legt das System diese soweit möglich aus den Eigenschaften des Kontexts fest, in den das Dokument eingefügt werden soll. Fehlende Parameter spezifiziert der Anwender (Abschnitt 6.4).

Systemunterstütztes Einfügen von Dokumenten:

Wie in Abschnitt 7.1.1 dargestellt, analysiert das System zum Modellierungszeitpunkt den Prozess und schlägt dem Bearbeiter eine Menge von Dokumenten vor, die als Hilfestellung für den Bearbeiter in unterschiedlichen Kontexten dienen und somit automatisch eingefügt werden können. Der Bearbeiter entscheidet für jedes Dokument, ob das Dokument in den vorgeschlagenen Kontext automatisch eingefügt werden soll.

Entfernen eines Kontextdokuments aus einem Kontext:

Liegt die entsprechende Berechtigung vor, kann der Anwender ein ausgewähltes Kontextdokument aus einem Kontext entfernen.

Suchen nach Kontextdokumenten:

Bei der Suche nach Kontextdokumenten spezifiziert der Anwender eine Menge von Bedingungen über die Attribute von Kontextdokumenten. Die entsprechende Treffermenge von Kontextdokumenten wird dem Anwender in einem separaten Fenster angezeigt.

Anzeigen und Modifizieren von Attributwerten von Kontextdokumenten:

Zu jedem ihm sichtbaren Kontextdokument kann sich der Anwender die Menge der Attribute und ihrer Werte anzeigen lassen. Liegt die entsprechende Berechtigung vor, kann der Anwender die Attributwerte modifizieren.

Notifikation des Dokumentverantwortlichen:

Ist der Gültigkeitszeitraum eines Kontextdokuments abgelaufen oder ist durch den Wert des Attributs „Revisionsrhythmus“ eine Aktualisierung des Dokuments notwendig, wird der entsprechende Dokumentverantwortliche durch das System benachrichtigt (Abschnitt 7.1.2).

8.2 Systemarchitektur

Aufbauend auf den beschriebenen Anwendungsfällen stellen wir im Folgenden einen Architekturvorschlag für die Erweiterung von WFMS um ein prozessbezogenes OMIS vor. Nach einer Übersicht über die Gesamtarchitektur erläutern wir die einzelnen Komponenten jeweils in einem separaten Abschnitt.

8.2.1 Übersicht

Bei der Konzeption der Systemarchitektur eines erweiterten WFMS, das die beschriebenen Eigenschaften von Kontexten umsetzt und somit für eine Unterstützung wissensintensiver Prozesse geeignet ist, wurden folgende Ziele verfolgt:

- Einsetzbarkeit für eine möglichst große Menge von existierenden WFMS:

Die vorgeschlagene Systemarchitektur erweitert nicht ein spezifisches Workflow-Management-Produkt. Vielmehr sollte durch wenige Annahmen über die Eigenschaften der zu erweiternden WFMS die Systemarchitektur für eine möglichst große Menge von WFMS einsetzbar sein. Die grundsätzlichen Voraussetzungen, die an ein WFMS gestellt werden, sind in Abschnitt 8.1.1 aufgeführt.

- Möglichst geringe Modifikationen an existierenden WFMS-Komponenten:

Die Erweiterung von WFMS sollte in der Form durchgeführt werden, dass die Funktionalität zur Visualisierung und zur Verwaltung von Kontexten möglichst in neu zu erstellenden Komponenten gekapselt ist und bestehende Komponenten des WFMS nur zu einem sehr geringen Anteil modifiziert werden. Insbesondere sollte der Workflow-Server nicht um neue Funktionalitäten erweitert werden müssen.

Abbildung 39 zeigt die Systemarchitektur eines zur Verwaltung von Kontexten erweiterten WFMS. Dabei bezeichnen wir mit dem Begriff „Workflow-Server“ die zentrale Komponente zur Verwaltung und Ausführung von Geschäftsfällen (entspricht gemäß Abbildung 1 dem Workflow Enaction Service in der WfMC-Architektur), und der Begriff „Workflow-Client“ bezeichnet die Komponente zur Verwaltung der Arbeitsliste bei der Durchführung von Prozessen (entspricht der Workflow Client Application in der WfMC-Architektur).

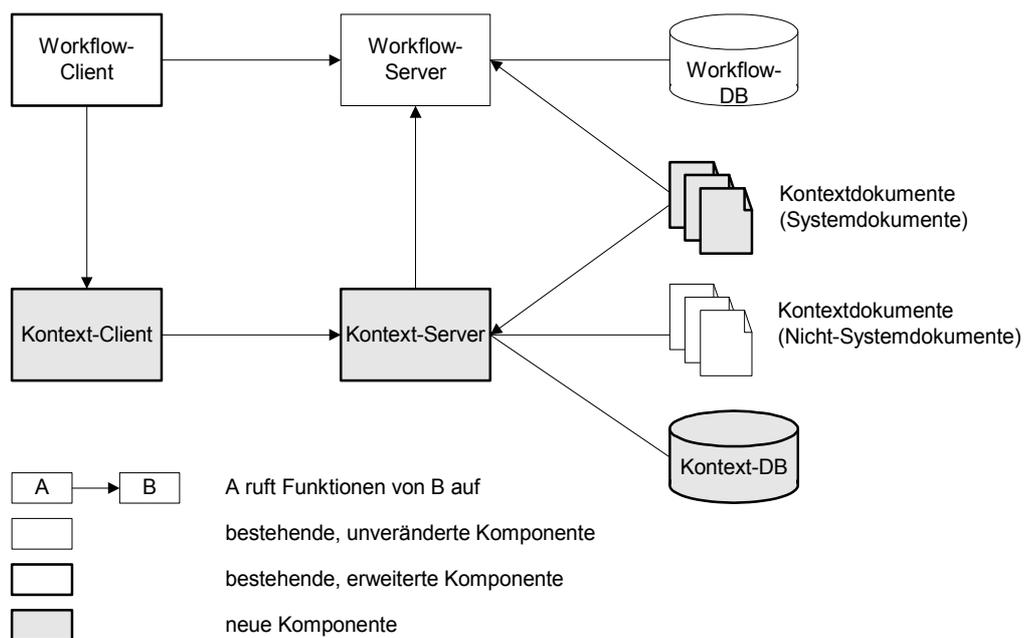


Abbildung 39: Architektur eines erweiterten WFMS

Die folgenden Erweiterungen ermöglichen die Verwaltung von Kontexten durch WFMS: Die zentrale Komponente stellt der Kontext-Server dar. Der Kontext-Server verwaltet alle Informationen zu Kontexten, kontextbildenden Objekten und Kontextdokumenten. Um die Kontextstruktur aufzubauen, kommuniziert der Kontext-Server mit dem Workflow-Server. Während die Attributwerte zu Kontextdokumenten und Kontexten in der Kontext-Datenbank gespeichert werden, verwaltet der Kontext-Server die in Form von Dateien vorliegenden Kontextdokumente im Dateisystem. Bestimmte Kontextdokumente, die Systemdokumente, prä-

sentieren dem Anwender Informationen des WFMS über z.B. laufende Geschäftsfälle. Dazu benötigen Systemdokumente den Zugriff auf entsprechende Funktionen des Workflow-Servers.

Der Workflow-Client, der geeignet erweitert werden muss, startet den Kontext-Client, falls der Anwender sekundäres Wissen benötigt. Darüber hinaus notifiziert der Workflow-Client den Kontext-Client über relevante Ereignisse der Prozessausführung, die mit Kontexten und Kontextdokumenten in Bezug stehen (Tabelle 13) und zum Anlegen neuer Kontexte oder zur automatischen Bereitstellung von Kontextdokumenten durch den Kontext-Client führen. Die Benutzerinteraktion zum Zugriff auf Kontexte und Kontextdokumente sowie die Visualisierung der Kontexte wird vom Kontext-Client unterstützt. Aus Gründen der Ergonomie sollten Workflow-Client und Kontext-Client in Bezug auf die Benutzerschnittstelle möglichst stark integriert sein.

In den folgenden Abschnitten werden die einzelnen Komponenten detaillierter erläutert. Dazu stellen wir zunächst das grundsätzliche Konzept sowohl der Visualisierung von Kontexten und Kontextdokumenten als auch der entsprechenden Benutzerinteraktion durch den Kontext-Client vor (Abschnitt 8.2.2). Die Integration des Kontext-Clients mit dem Workflow-Client und notwendige Erweiterungen des Workflow-Clients werden in Abschnitt 8.2.3 beschrieben. Darauf aufbauend geben wir einen Überblick über die Funktionen, die der Kontext-Server an der Schnittstelle bereitstellen muss (Abschnitt 8.2.4), und wir erläutern den Aufbau der Kontext-Datenbank (Abschnitt 8.2.5). Die Integration von Systemdokumenten in die beschriebene Architektur wird in Abschnitt 8.2.6 dargestellt. Eine Modifikation des Workflow-Servers muss für den vorliegenden Architekturvorschlag nicht vorgenommen werden. Daher beschreiben wir abschließend in Abschnitt 8.2.7 lediglich, welche funktionalen Anforderungen in Bezug auf die Verwaltung von Kontexten an den Workflow-Server gestellt werden.

8.2.2 Kontext-Client

Durch den Kontext-Client werden dem Anwender die Struktur der Kontexte und die entsprechenden Kontextdokumente visualisiert. Darüber hinaus bietet der Kontext-Client die bei der Diskussion der Anwendungsfälle aufgeführten Funktionen auf Kontexten und Kontextdokumenten an. Der grundsätzliche Aufbau des Kontext-Clients wird in Abbildung 40 skizziert.

Der Kontext-Client ist logisch in drei Bereiche aufgeteilt: Das Funktionsmenü stellt die oben genannten Funktionen auf Kontexten und Kontextdokumenten bereit. Der Navigationsbereich visualisiert die Kontextstruktur in graphischer Form und erlaubt die Navigation über die Menge der Kontexte und die Auswahl eines Kontexts. Die Kontextdokumente des ausgewählten Kontexts werden in der Dokumentenliste dargestellt. Die einzelnen Bereiche des Kontext-Clients werden im Folgenden detaillierter erläutert:

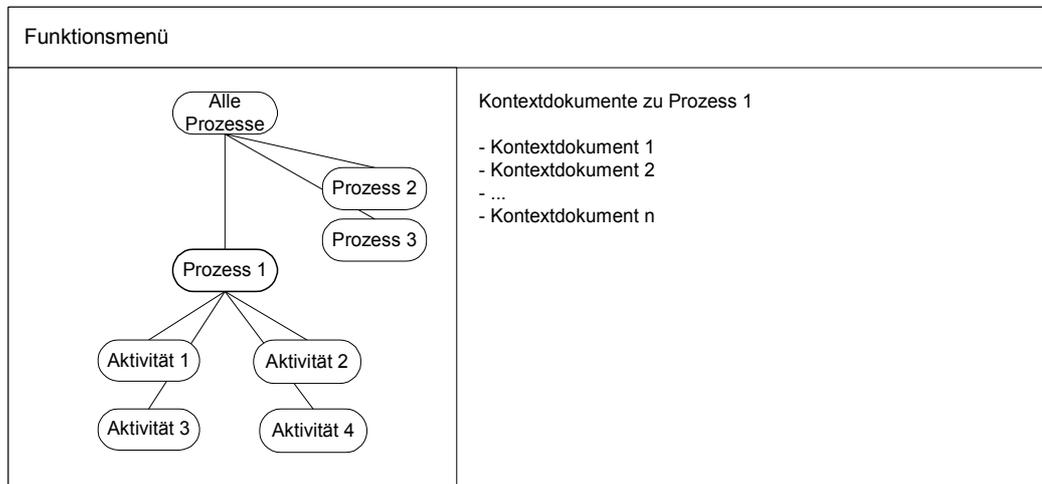


Abbildung 40: Aufbau des Kontext-Clients (Beispiel)

Die Navigationskomponente stellt zwei unterschiedliche Modi bereit, die einerseits dem Bearbeiter die freie Navigation innerhalb der Kontextmenge gewähren und andererseits den direkten Zugriff auf die Kontexte erlauben, die bei der Durchführung einer konkreten Aktivität eines Geschäftsfalles die größte Relevanz besitzen. Diese Kontexte werden unter dem Begriff „aktueller Kontext“ zusammengefasst.

Freie Navigation innerhalb der Kontextmenge:

Die Kontextstruktur muss für den Bearbeiter so dargestellt werden, dass eine einfache Navigation innerhalb der Kontextmenge möglich ist. Daher wird an eine Umsetzung die Anforderung gestellt, dass die in Abbildung 32 dargestellten Objekttypen und die zwischen den Objekttypen bestehenden Beziehungen in der Navigationsstruktur dargestellt werden und somit als Navigationspfade für den Bearbeiter verfügbar sind. Dies bedingt beispielsweise, dass der Bearbeiter ausgehend von einem Prozess-Kontext direkt zu den zugehörigen Aktivitäts-Kontexten navigieren kann oder dass die hierarchische Struktur der Organisationseinheiten mit den jeweiligen Bearbeitern und Rollen in der Navigationskomponente umgesetzt ist. Zusätzlich sollten verschiedene Startpunkte in die Kontextstruktur verfügbar sein, um auf einfache Weise z.B. alle Prozesse, alle Organisationseinheiten oder alle Werkzeuge anzeigen lassen zu können. Die notwendigen Informationen über die existierenden Kontexte werden vom Kontext-Server bereitgestellt.

Die Visualisierung der Kontextmenge durch eine Baumstruktur wie bei klassischen Werkzeugen zur Darstellung von Verzeichnisstrukturen (z.B. Microsoft Windows Explorer) führt nur bei einer geringen Menge von kontextbildenden Objekttypen zu übersichtlichen Ergebnissen. Daher soll der Navigationsbereich durch eine Komponente realisiert werden, die die Beziehungen zwischen kontextbildenden Objekten (und somit den Kontexten) auf graphische Weise visualisiert, ohne eine Baumstruktur vorzugeben. Wir werden im folgenden Kapitel diesbezüglich zwei Implementierungen vorstellen. Der erste Prototyp WoMIS (Abschnitt 9.2), der nur

eine sehr eingeschränkte Menge von kontextbildenden Objekttypen, nämlich „Prozess“, „Aktivität“ und „Geschäftsfall“, unterstützt, visualisiert die Kontexte in einer Baumstruktur ähnlich dem MS Windows Explorer. Dagegen nutzt der erweiterte Prototyp KontextNavigator (Abschnitt 9.3) eine Darstellung, die sich an der in Abbildung 40 skizzierten Visualisierung und Navigation auf Kontexten orientiert.

Direkter Zugriff auf den aktuellen Kontext:

Führt der Bearbeiter eine konkrete Aktivität in einem Geschäftsfall durch, benötigt er eine Unterstützung durch das System, die über die freie Navigation in der Menge der Kontexte hinausgeht. Die Bereitstellung sekundären Wissens, das für den Bearbeiter in der aktuellen Bearbeitungssituation von besonderer Relevanz ist, wird durch den „aktuellen Kontext“ unterstützt. Dabei umfasst der aktuelle Kontext mehrere Kontexte, die einen besonders starken Bezug zur aktuellen Bearbeitung besitzen. Es sind dies

- der Kontext des Geschäftsfalls, den der Bearbeiter gerade durchführt,
- der Kontext der Aktivität (im aktuellen Geschäftsfall), die der Bearbeiter gerade durchführt,
- der Kontext der aktuellen Aktivitätsausführung (d.h. der Kontext, der an beide der oben genannten Objekte gebunden ist und die Ausführung der Aktivität in diesem einen Geschäftsfalls unterstützt) und
- der Kontext zum Prozess, dem der Geschäftsfall zugeordnet ist.

Um den Zugriff auf die Kontextdokumente des aktuellen Kontexts zu erleichtern, zeigt die Dokumentenliste bei der Auswahl des aktuellen Kontexts die Kontextdokumente dieser vier Kontexte gemeinsam an.

Der aktuelle Kontext wird in die oben beschriebene Navigationsstruktur von Kontexten integriert. Dazu werden die Beziehungen zu den vier Kontexten, aus denen sich der aktuelle Kontext zusammensetzt, graphisch dargestellt. Darüber hinaus werden auch die Beziehungen zu weiteren Kontexten visualisiert, wie zum Kontext des Bearbeiters, zum Kontext der Rolle, die der Bearbeiter gerade innehat, oder zum Kontext der Organisationseinheit, zu der der Bearbeiter zugehörig ist. Die Navigation zu beliebigen Kontexten gemäß der oben skizzierten Navigationsstruktur ausgehend vom aktuellen Kontext ist daher auf einfache Weise möglich. Beispielsweise kann der Bearbeiter vom aktuellen Kontext aus direkt in den Kontext des aktuell zu bearbeitenden Geschäftsfalls navigieren und von dort aus auf einfache Weise Kontexte anderer Geschäftsfälle des Prozesses anwählen. Abbildung 41 skizziert die Darstellung des aktuellen Kontexts im Kontext-Client.

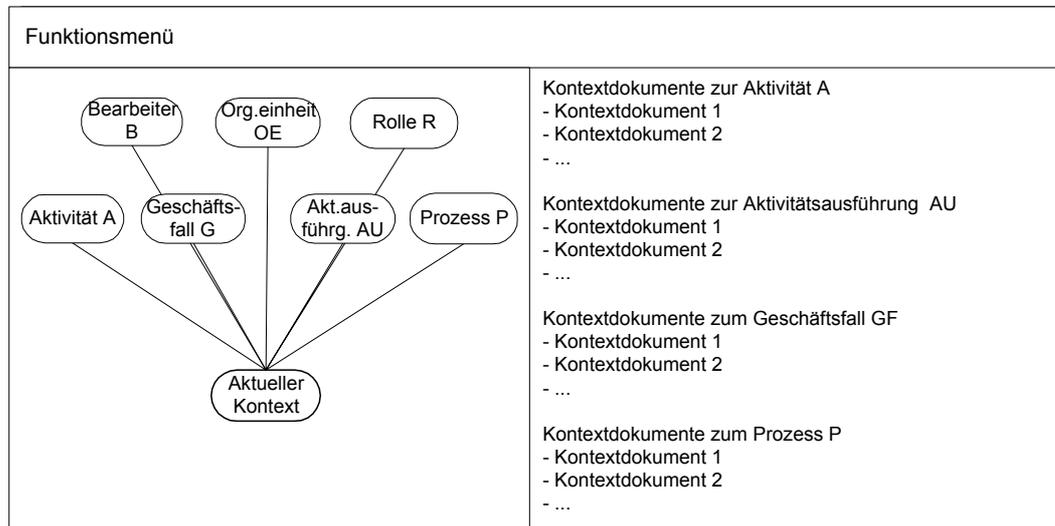


Abbildung 41: Darstellung des aktuellen Kontexts

Die Dokumentenliste zeigt die Kontextdokumente zum ausgewählten Kontext jeweils zusammen mit ausgewählten Attributwerten an. Über das Funktionsmenü stehen dem Anwender die in Abschnitt 8.1.2 aufgeführten Funktionen zu Kontextdokumenten zur Verfügung. Bei der Ausführung dieser Funktionen kommuniziert der Kontext-Client mit dem Kontext-Server.

Bestimmte Funktionalitäten des Kontext-Clients sollen lediglich dem Administrator bzw. dem Modellierer zur Verfügung stehen. Es handelt sich dabei um die folgenden Operationen auf Kontexten und Kontextdokumenten:

- Kontexte anlegen:

Kontexte zu Werkzeugen und Aktivitätsblöcken können vom Modellierer manuell erzeugt werden. Der Kontext-Client muss also insbesondere Funktionalitäten enthalten, über die bestehende Aktivitäten zu einem Aktivitätsblock gruppiert werden können.

- Kontexte löschen:

Das Löschen von Kontexten erfolgt ausschließlich durch den Modellierer. Somit kann auch auf Erfahrungen z.B. zurückliegender Geschäftsfälle zurückgegriffen werden, bis diese Kontexte ausdrücklich gelöscht werden. Darüber hinaus ist diese Funktionalität notwendig, wenn im Rahmen der kontinuierlichen Verbesserung von Prozessen z.B. Aktivitäten aus dem Prozess entfernt werden. Der Modellierer kann daraufhin die Kontextdokumente der zugehörigen Aktivitäts-Kontexte jeweils auf ihre Relevanz bewerten, diese ggf. anderen Kontexten zuordnen und schließlich die Aktivitäts-Kontexte löschen.

- Systemunterstütztes Einfügen von Kontextdokumenten:

Wie in Abschnitt 7.1.1 beschrieben, schlägt das System dem Bearbeiter eine Liste von Kontextdokumenten vor, die initial automatisch in entsprechende Kontexte eingefügt werden können. Diese Liste von Dokumenten wird vom Kontext-Server bereitgestellt. Der Modellierer entscheidet für jedes einzelne Dokument, ob das System das Dokument in den zugehörigen Kontext einfügen soll.

8.2.3 Workflow-Client

Um die Verwaltung von Kontexten auf der Ebene der Benutzerinteraktion in eine WFMS-Architektur integrieren zu können, muss der Workflow-Client in Bezug auf die folgenden beiden Aspekte erweitert werden:

- Start des Kontext-Clients bzw. Wechsel des dargestellten aktuellen Kontexts im Kontext-Client aus dem Workflow-Client heraus
- Benachrichtigung des Kontext-Clients über relevante Ereignisse der Geschäftsfallbearbeitung

Die Durchführung von Geschäftsfällen und die Bearbeitung von Prozessmodellldokumenten erfolgt durch den Workflow-Client. Handelt es sich um einen wissensintensiven Prozess, der zur Bearbeitung ansteht, und benötigt der Bearbeiter sekundäres Wissen, startet er den Kontext-Client aus dem Workflow-Client heraus. Dabei sind beide Modi der Darstellung von Kontexten durch den Kontext-Client geeignet zu integrieren.

Der Workflow-Client sollte erstens um einen Menüpunkt erweitert werden, über den der Bearbeiter den Kontext-Client im Modus „Freie Navigation“ starten kann. Darüber hinaus sollte der erweiterte Workflow-Client zulassen, dass bei der Durchführung eines Geschäftsfalls der Bearbeiter den Kontext-Client im Modus „Aktueller Kontext“ starten kann (falls der Kontext-Client bereits gestartet wurde, wechselt die Anzeige entsprechend in den neuen aktuellen Kontext). Die exakte Integration von entsprechenden Funktionalitäten hängt vom Design des Workflow-Clients ab. Zumeist visualisieren Workflow-Clients eine Arbeitsliste, d.h. eine Liste von zur Bearbeitung anstehender Geschäftsfälle. Eine Erweiterung des Workflow-Clients sollte dann folgende Funktionalität enthalten: Wählt der Bearbeiter einen Geschäftsfall aus, wird ein Menüeintrag oder eine Schaltfläche angezeigt, deren Aktivierung den Aufruf des Kontext-Clients im Modus „Aktueller Kontext“ durchführt. Die in Kapitel 9 vorgestellten Prototypen zeigen jeweils eine mögliche Integration anhand eines auf Internet-Technologien basierenden Workflow-Clients.

Darüber hinaus benachrichtigt der Workflow-Client den Kontext-Client über Ereignisse, die einerseits zum Anlegen neuer Kontexte führen (Start eines Geschäftsfalls) oder zur automatischen Bereitstellung von Kontextdokumenten durch das System führen können. Die Menge der diesbezüglich relevanten Ereignisse ist in Tabelle 13 aufgeführt. Der Kontext-Client reagiert auf diese Ereignisse, indem er die entsprechenden Funktionen des Kontext-Servers aufruft.

8.2.4 Kontext-Server

Der Kontext-Server ist für die Verwaltung von Kontexten und Kontextdokumenten zuständig. Dazu stellt er eine Menge von Funktionen bereit, die vom Kontext-Client aufgerufen werden. Die einzelnen Funktionen, die aus den Funktionalitäten des Kontext-Clients abgeleitet werden können, werden hier nicht auf der Ebene einer exakten durch Funktionsname und Parameter spezifizierten Schnittstellendefinition vorgestellt. Wir geben vielmehr eine informelle Beschreibung der Funktionen, die der Kontext-Server an der Schnittstelle bereitstellen muss. Diese informelle Beschreibung reicht aus, um die Funktionalität des Kontext-Servers zu erläutern. Zusätzlich stellen wir zu den einzelnen Funktionen dar, welche Daten bei der Ausführung der Funktion mit weiteren Komponenten der Architektur ausgetauscht werden bzw. welche Schnittstellenfunktionen weiterer Komponenten aufgerufen werden.

Funktionen zur Verwaltung von Kontexten:

Der Kontext-Server stellt sicher, dass der Kontext-Client die Kontextstruktur im Navigator darstellen kann. Die Kontextstruktur ergibt sich zum überwiegenden Teil aus den kontextbildenden Objekten des WFMS. Darüber hinaus kann der Modellierer manuell bestimmte kontextbildende Objekte (und damit Kontexte) erzeugen. Der dynamische Zugriff auf die gesamte Menge der Kontexte zur Darstellung im Kontext-Client kann prinzipiell auf zwei unterschiedliche Arten realisiert werden:

- Der Kontext-Server greift auf die kontextbildenden Objekte, die vom WFMS verwaltet werden, über die Schnittstelle des Workflow-Servers zu, der die entsprechenden Informationen aus der Workflow-Datenbank ausliest und an den Kontext-Server übergibt. Die Informationen über manuell angelegte kontextbildende Objekte werden vom Kontext-Server dagegen in der Kontext-Datenbank verwaltet und aus dieser ausgelesen.
- Die relevanten Attributwerte der kontextbildenden Objekte, die vom WFMS verwaltet werden, werden als Kopie in der Kontext-Datenbank abgelegt und sind somit zusammen mit den Informationen über manuell angelegte kontextbildende Objekte durch den Kontext-Server zugreifbar, da letztere ebenfalls in der Kontext-Datenbank gespeichert werden.

Für das Design des Kontext-Servers und der Kontext-Datenbank wurde die zweite Alternative ausgewählt: Die einheitliche Verwaltung der kontextbildenden Objekte, zwischen denen verschiedene Beziehungen bestehen (Abbildung 42), wird durch die einheitliche Speicherung in einer Datenbank erleichtert. Insbesondere sind Performanzprobleme des Gesamtsystems absehbar, falls in der ersten Alternative jeder Zugriff auf die Kontextstruktur über ggf. eine Vielzahl von Funktionsaufrufen des Workflow-Servers abgewickelt werden muss.

Zur Verwaltung von Kontexten muss der Kontext-Server die folgenden Funktionen bereitstellen:

- **Erzeuge initiale Kontextstruktur:**

Diese Funktion wird beim Start des Kontext-Servers aufgerufen und transferiert die relevanten Attributwerte der kontextbildenden Objekte des WFMS in die Kontext-Datenbank. Dazu werden entsprechende Funktionen des Workflow-Servers aufgerufen, die diese Informationen liefern. Die Implementierung dieser Funktion ist daher abhängig vom konkreten WFMS.

- **Erzeuge Kontext:**

Diese Funktion wird einerseits aufgerufen, falls der Modellierer einen Kontext, z.B. zu einem Werkzeug, manuell anlegt. Andererseits dient diese Funktion zum Anlegen von Geschäftsfall-Kontexten, falls der Bearbeiter im Workflow-Client einen neuen Geschäftsfall erzeugt. Der Workflow-Client benachrichtigt dazu den Kontext-Client über dieses Ereignis, der daraufhin diese Funktion aufruft. Wie auch die folgenden beiden Funktionen greift diese Funktion lediglich auf die Kontext-Datenbank zu.

- **Lösche Kontext:**

Durch diese Funktion werden ein kontextbildendes Objekt und alle Kontextdokumente des zugehörigen Kontexts aus der Kontext-Datenbank gelöscht.

- **Lese Kontextstruktur:**

Der Kontext-Server liefert durch diese Funktion die Menge der Kontexte (oder ggf. einen Ausschnitt).

Funktionen zur Verwaltung von Kontextdokumenten:

Abschnitt 6.3 beschreibt den Aufbau des Informationsmodells für Kontextdokumente. Die Attributwerte von Kontextdokumenten werden jeweils in der Kontext-Datenbank gespeichert. Die folgenden Funktionen bieten dem Kontext-Client den Zugriff auf diese Attributwerte, um z.B. die Dokumentenliste zu einem Kontext aufzubauen sowie Kontextdokumente zu speichern oder zu löschen.

- **Erzeuge Kontextdokument:**

Diese Funktion legt in der Kontext-Datenbank ein Kontextdokument mit den übergebenen Attributwerten an.

- **Ändere Kontextdokument:**

Bei der Ausführung dieser Funktion prüft der Kontext-Server vorab, ob der Bearbeiter die entsprechende Berechtigung zur Änderung der Attributwerte eines Kontextdokuments besitzt.

- **Lösche Kontextdokument:**

Bei der Ausführung dieser Funktion prüft der Kontext-Server vorab, ob der Bearbeiter die entsprechende Berechtigung zum Löschen eines Kontextdokuments besitzt.

- **Lese Kontextdokument:**

Der Kontext-Server liefert ohne weitere Prüfungen die entsprechenden Attributwerte zum angegebenen Kontextdokument zurück.

- Lese Kontextdokumente zu einem Kontext:

Diese Funktion liefert die Informationen zum Aufbau der Dokumentenliste im Kontext-Client. Dabei wird zunächst aus der Kontextdatenbank die Menge der Kontextdokumente zum angegebenen Kontext abgefragt, und es werden für jedes Kontextdokument daraufhin die folgenden Abfragen durchgeführt:

Zunächst wird getestet, ob der Bearbeiter die Berechtigung zur Anzeige des Dokuments besitzt: Ist die das Attribut „Eingeschränkt sichtbar“ für das Kontextdokument gesetzt, wird das Dokument nur den in der Adressatengruppe angegebenen Bearbeitern angezeigt. Daraufhin wird überprüft, ob das aktuelle Datum im Gültigkeitszeitraum des Dokuments liegt. Schließlich wird untersucht, ob ggf. definierte prozessbezogene Sichtbarkeitsbedingungen erfüllt sind. Der Kontext-Server fragt zu diesem Zweck jeweils die Werte der zur Prüfung einer Sichtbarkeitsbedingung relevanten Variablen (wie z.B. „Priorität“ des aktuellen Geschäftsfalls oder „Discount“, der dem Kunden im aktuellen Geschäftsfall gewährt wird) vom Workflow-Server ab. Es werden nur die Attributwerte zu den Kontextdokumenten, bei denen alle drei Kriterien erfüllt sind, an den Kontext-Client zurückgegeben.

- Lese aktiv bereitzustellende Kontextdokumente:

Diese Funktion wird vom Kontext-Client aufgerufen und liefert die Menge der Attributwerte der Kontextdokumente zurück, die der Kontext-Client beim Auftreten eines relevanten Ereignisses dem Bearbeiter aktiv bereitstellen soll. Der Kontext-Client liefert dazu das entsprechende Ereignis, woraufhin der Kontext-Server die entsprechenden Kontextdokumente identifiziert. Die Auswahl der aktiv bereitzustellenden Kontextdokumente erfolgt gemäß Abbildung 35.

- Suche Kontextdokument:

Die Kontext-Datenbank wird gemäß den spezifizierten Kriterien durchsucht und die entsprechende Treffermenge von Kontextdokumenten zurückgegeben.

Funktionen zu Dateien:

- Speichere Datei:
- Lösche Datei:
- Lese Datei:

Kontextdokumente, die in Form von Dateien vorliegen, werden vom Kontext-Server außerhalb der Kontext-Datenbank gespeichert. Wird ein Kontextdokument in einen Kontext eingefügt, das auf eine WWW-Seite verweist, werden vom Kontext-Server lediglich die entsprechenden Attributwerte zu diesem Kontextdokument angelegt. Handelt es sich aber um ein dateibasiertes Kontextdokument, wie z.B. ein Word-Dokument, wird zusätzlich die Funktion „Speichere Datei“ aufgerufen, die die Datei an den Kontext-Server übergibt. Eine analoge Vorgehensweise ergibt sich beim Löschen oder beim Lesen von dateibasierten Kontextdokumenten.

Weitere Funktionen:

- Lese Menge von initialen Kontextdokumenten:

Diese Funktion liefert eine Menge von Kontextdokumenten zurück, die dem Anwender als automatisch einzufügende Kontextdokumente angeboten werden. Der Kontext-Server greift dazu zunächst auf die kontextbildenden Objekte in der Kontextdatenbank zu und erstellt entsprechend der Beschreibung in Abschnitt 7.1.1 eine Liste von potentiellen Kontextdokumenten für die einzelnen Kontexte. Dabei handelt es sich um Systemdokumente der Komplexitätsstufe 1 (Informationen zu einem Prozess, zu einer Aktivität, zu einer Dokumentvorlage, etc.). Um weitere potentielle Kontextdokumente zu identifizieren (Kontextdokumente zur Unterstützung von Entscheidungen), muss der Kontext-Server die Struktur des Prozesses analysieren. Die entsprechenden Informationen fragt der Kontext-Server über die Schnittstelle des Workflow-Servers ab. Die Funktion liefert schließlich die gesamte Menge von möglichen Kontextdokumenten an den Kontext-Client zurück.

- Notifiziere Kontextverantwortlichen:

Wie in Abschnitt 7.1.2 geschildert, unterrichtet das System den Dokumentverantwortlichen eines Kontextdokuments, falls der für das Dokument angegebene Gültigkeitszeitraum überschritten wurde oder das im Attribut „Revisionsrhythmus“ spezifizierte Zeitintervall abgelaufen ist. Diese Bedingungen werden von dieser Funktion einmal täglich automatisch für den gesamten Bestand von Kontextdokumenten überprüft, und es wird die Notifikation der entsprechenden Kontextverantwortlichen per E-Mail vorgenommen. Es handelt sich hier nicht um eine Funktion, die an der Schnittstelle des Kontext-Servers angeboten wird, sondern um einen ständig laufenden Prozess des Kontext-Servers.

8.2.5 Kontextdatenbank

Abschnitt 42 zeigt den Aufbau der Kontextdatenbank als Entity-Relationship-Diagramm. Werden die existierenden many-to-many-Beziehungen zwischen Entities aufgelöst, kann auf einfache Weise eine vollständige Tabellenstruktur für ein relationales Datenbanksystem abgeleitet werden.

Im linken Teil der Abbildung sind die Entities dargestellt, die die kontextbildenden Objekte vom Typ „Bearbeiter“, „Rolle“, „Organisationseinheit“, „Aktivitätsblock“, „Aktivität“, „Prozess“, „Geschäftsfall“, „Dokumentvorlage“ und „Werkzeug“ repräsentieren. Jedes dieser Entities enthält als Attribute den Identifikator des entsprechenden Objekts im WFMS, den Bezeichner, eine Beschreibung und ggf. weitere beschreibende Attribute des Objekts im WFMS, die genutzt werden, um dem Anwender relevante Informationen über das Objekt im Kontext-Client anzuzeigen.

Das Entity „Kontextdokument“ stellt die kontextbezogenen Attribute dar, während das Entity „Dokument“ die dokumentbezogenen Attribute repräsentiert. Die entsprechenden Attribute sind in Abschnitt 6.3 aufgeführt (wobei die Attribute „Relevanz“, „Stichworte“, „Sichtbarkeit“, „Ereignisse“ und die Attribute des Aspekts „Berechtigungen“ in der Kontextdatenbank durch eigene Entities repräsentiert sind). Kontextdokumente können auf weitere Kontextdokumente verweisen, daher besteht für das Entity „Kontextdokument“ eine rekursive many-to-many-Beziehung.

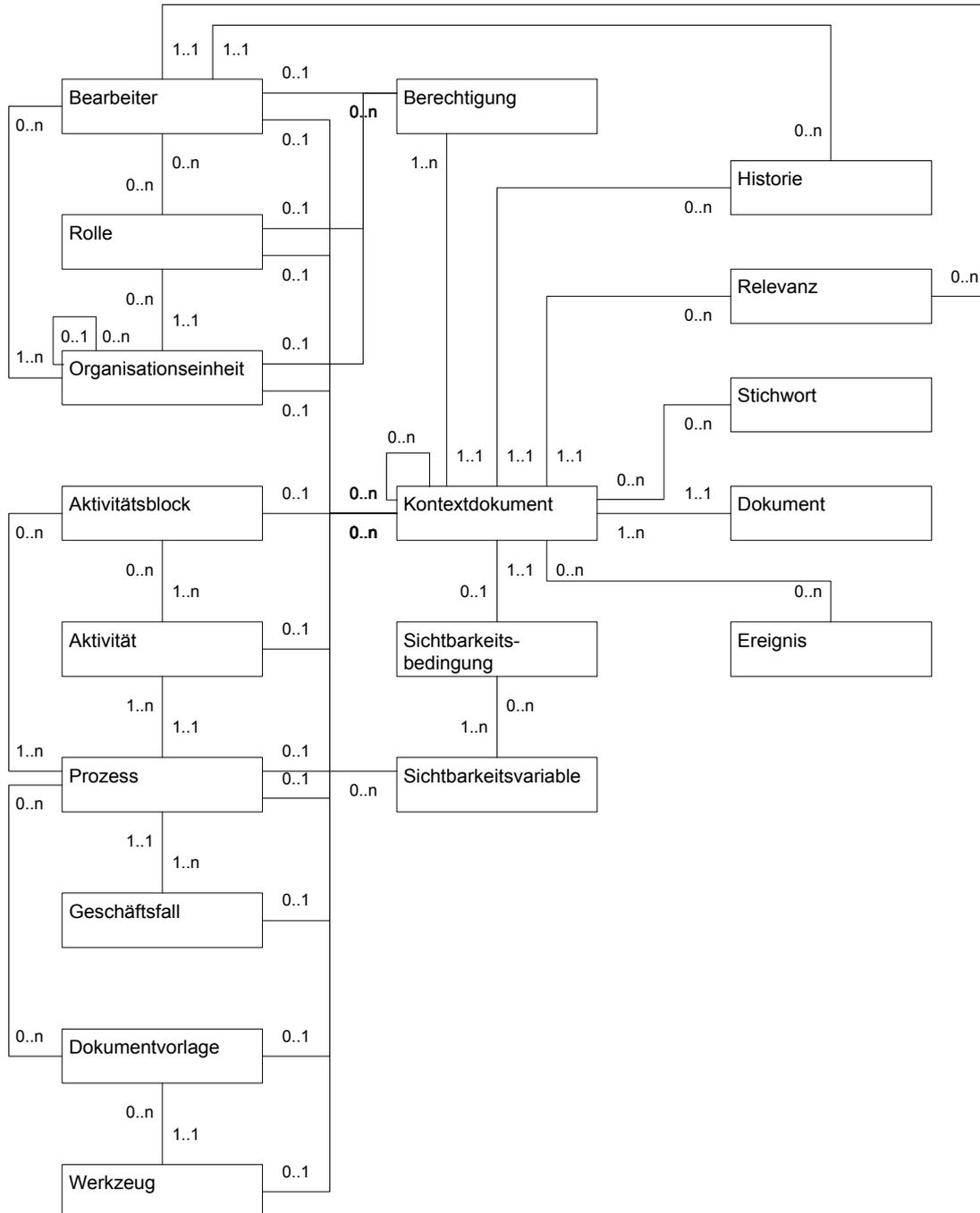


Abbildung 42: Aufbau der Kontextdatenbank

Die folgenden Entities dienen zur weiteren Charakterisierung eines Kontextdokuments (die exakten Attribute der Entities sind aus der Beschreibung des Informationsmodells in Abschnitt 6.3 ablesbar): Die Berechtigungen, die Bearbeiter, Rollen oder Organisationseinheiten auf einem Kontextdokument besitzen, werden durch das Entity „Berechtigung“ repräsentiert. Die Historie des Zugriffs auf das Kontextdokument wird durch das Entity „Historie“ dargestellt, und die Einschätzung der Relevanz des Kontextdokuments durch die Bearbeiter findet sich in dem Entity „Relevanz“ wieder. Das Entity „Stichwort“ repräsentiert die Liste der Stichworte, die einem Kontextdokument zugeordnet werden können.

Das Entity „Sichtbarkeitsvariable“ beschreibt die Menge der prozessbezogenen Variablen, die zur Formulierung von Sichtbarkeitsbedingungen verwendet werden können. Dabei handelt es sich sowohl um systemspezifische Variablen zu Geschäftsfällen und Aktivitätsausführungen, die vom Administrator des Systems initial festgelegt werden als auch um prozessspezifische Variablen zu Geschäftsfällen, die vom Modellierer für jeden Prozess spezifiziert werden (Abschnitt 6.3.5 und Abbildung 34). Handelt es sich um eine Variable, die nur für einen bestimmten Prozess gültig ist (also z.B. „Discount“ für den Prozess „Contract Management“), gibt die Beziehung zum Entity „Prozess“ den entsprechenden Prozess an. Über das Entity „Sichtbarkeitsbedingung“ werden mit Hilfe der Sichtbarkeitsvariablen konkrete Sichtbarkeitsbedingungen für ein Kontextdokument festgelegt (siehe auch hier Abschnitt 6.3.5 bzgl. der Syntax von Sichtbarkeitsbedingungen).

8.2.6 Systemdokumente

Systemdokumente werden durch Funktionalitäten des Kontext-Clients aus der Standardbibliothek von Systemdokumenttypen erzeugt und in einen Kontext eingefügt. Wird ein Systemdokument erzeugt, werden bestimmte Parameter, die aus dem Kontext abgeleitet werden können, im Systemdokument gespeichert. Wird z.B. in einem Prozesskontext zum Prozess „Contract Management“ ein Systemdokument vom Typ „Informationen zu allen laufenden und abgeschlossenen Geschäftsfällen eines Prozesses“ instanziiert, speichert das System den Identifikator des Prozesses direkt im Systemdokument. Das Systemdokument zeigt daraufhin bei jedem Öffnen die aktuellen Informationen zu Geschäftsfällen des Prozesses „Contract Management“ an.

Darüber hinaus kann der Bearbeiter bei jedem Öffnen des Systemdokuments die Menge der Geschäftsfälle, die angezeigt werden sollen, über die Formulierung von Bedingungen einschränken. Das Systemdokument muss daher dem Anwender direkt bei jedem Öffnen eine entsprechende Maske präsentieren.

Mit Hilfe der fest im Systemdokument gespeicherten Parameter (hier: Prozessidentifikator) und der bei jedem Öffnen des Systemdokuments spezifizierten Parameter (Einschränkung der anzuzeigenden Geschäftsfallmenge) ruft das Systemdokument geeignete Funktionen des Workflow-Servers auf, der die entsprechenden Informationen zurückliefert. Systemdokumente sind schreibgeschützt und können somit vom Anwender nicht modifiziert werden.

8.2.7 Workflow-Server

Der Workflow-Server soll ohne Modifikation in die erweiterte WFMS-Architektur integriert werden können. Daher ist an dieser Stelle nicht die Beschreibung einer funktionalen Erweiterung notwendig, sondern es soll lediglich skizziert werden, welche Funktionen der Workflow-Server an seiner Schnittstelle zum Aufruf durch den Kontext-Server bereitstellen muss. Es sind dies die folgenden drei Klassen von Funktionen:

- Funktionen zum Lesen der Attributwerte von kontextbildenden Objekten:
Die Funktion „Erzeuge initiale Kontextstruktur“ des Kontext-Servers fragt vom Workflow-Server die Attributwerte zu den im WFMS gespeicherten kontextbildenden Objekten ab. Der Workflow-Server muss also die entsprechenden Funktionen zum Zugriff auf diese Informationen anbieten.
- Funktionen zum Lesen von Variablenwerten zur Auswertung von Sichtbarkeitsbedingungen:
Zur Auswertung von Sichtbarkeitsbedingungen auf Kontextdokumenten, wie z.B. „Zeige das Dokument nur an, wenn die Priorität des aktuell bearbeiteten Geschäftsfalls = 1 ist“, benötigt der Kontext-Server Zugriff auf die Werte der entsprechenden Variablen (hier „Priorität“) von Geschäftsfällen oder Aktivitäten. Diese müssen über Funktionen des Workflow-Servers abfragbar sein.
- Funktionen zum Lesen der Informationen, die in Systemdokumenten dargestellt werden:
Der Workflow-Server muss Funktionen anbieten, die die in Systemdokumenten anzuzeigenden Informationen bereitstellen.

Die Funktionen, die der Workflow-Server bereitstellen muss, bieten lediglich den lesenden, nicht-modifizierenden Zugriff auf Daten der Workflow-Datenbank. Sollte der Workflow-Server bestimmte hier geforderte Funktionen nicht anbieten, kann alternativ der Kontext-Server durch direkten Zugriff auf die Workflow-Datenbank die entsprechenden Informationen auslesen.

9 Realisierung

In diesem Kapitel stellen wir zwei Prototypen vor, die zusammen einen Großteil der in dieser Arbeit vorgestellten Konzepte realisieren. Dabei handelt es sich bei dem System WoMIS (Workflow Memory Information System) um einen im Rahmen des MOVE-Projekts entstandenen frühen Prototypen, der in vielen Bereichen nur die Basismechanismen von Kontexten umsetzt. Wir stellen daher die wichtigsten Aspekte von WoMIS in Abschnitt 9.2 nur kurz vor. Das zweite prototypische System KontextNavigator realisiert dagegen den überwiegenden Teil der in dieser Arbeit vorgestellten Konzepte und wird daher in Abschnitt 9.3 in größerer Breite behandelt. Beide Prototypen stellen eine Erweiterung des WFMS CSE Workflow dar, dessen softwaretechnische Architektur im folgenden Abschnitt zunächst kurz skizziert wird.

9.1 Die Architektur des Workflow-Management-Systems CSE Workflow

Abbildung 43 zeigt die Architektur des WFMS CSE Workflow (Abschnitt 2.1.3) in der Version „Gold Edition“. Neben den hier nicht relevanten Clients zur Organisations- und Prozessmodellierung stellt CSE Workflow zwei unterschiedliche Workflow-Clients zur Durchführung von Geschäftsfällen bereit. Dabei bietet das System neben dem Client „WorkDesk“, der direkt auf den Workflow-Server zugreift, einen Client „WebDesk“, der auf Internet-Technologien basiert [CSE99b]. Die beiden im Rahmen dieser Arbeit erstellten Prototypen nutzen letztere Komponente, die daher im Folgenden kurz skizziert wird.

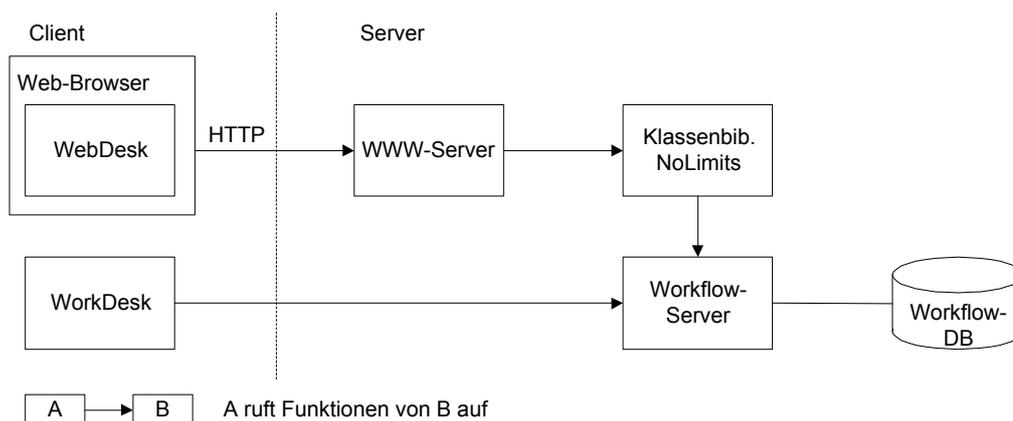


Abbildung 43: Die Architektur des WFMS CSE Workflow

WebDesk bietet nicht die volle Funktionalität des WorkDesk. Die wichtigsten Funktionen zur Bearbeitung von Geschäftsfällen können allerdings mit WebDesk durchgeführt werden (Abbildung 46). Dabei handelt es sich um:

- die Authentifizierung eines Bearbeiters,

- die Ansicht der Arbeitsliste, d.h. aller Geschäftsfälle, die einem Bearbeiter zur Durchführung zugewiesen wurden,
- die Ansicht der Attributwerte eines Geschäftsfalls,
- die Bearbeitung eines Geschäftsfalls durch Ansicht und Modifikation der elektronischen Umlaufmappe (z.B. durch Einfügen neuer Dokumente oder Bearbeitung vorhandener Dokumente),
- das Erzeugen von neuen Geschäftsfällen und
- die Suche nach Geschäftsfällen.

WebDesk besteht aus einer Menge von HTML-Seiten, die auf dem Client-Rechner in einem WWW-Browser angezeigt werden. Um einen dynamischen Zugriff auf die vom Workflow-Server in der Workflow-Datenbank verwalteten Informationen sicherzustellen, verwendet WebDesk die JAVA-Klassenbibliothek NoLimits [CSE99a]. Die NoLimits-Klassen erlauben den lesenden und schreibenden Zugriff auf die Workflow-Datenbank über den Workflow-Server und erzeugen dynamisch die im WWW-Browser angezeigten HTML-Seiten.

9.2 Das prototypische System Workflow Memory Information System (WoMIS)

Bei WoMIS (Workflow Memory Information System) handelt es sich um einen im Rahmen des MOVE-Projekts entstandenen frühen Prototypen, der daher das in Kapitel 6 und Kapitel 7 beschriebene Konzept der Kontexte und deren organisatorische Einbettung nur zu einem Teil umsetzt. Dennoch enthält WoMIS bereits die relevanten Basiseigenschaften von Kontexten und zeigt deren Integration in das WFMS CSE Workflow. Das System wurde auf dem Abschlusskongress des MOVE-Projekts vorgestellt, und weitere Informationen finden sich in [DGL00, GH00, GH01 und Goes00].

9.2.1 Funktionalität

Bei WoMIS handelt es sich um einen Prototypen, durch den das Grundkonzept von Kontexten und ihre Integration in die Workflow-Bearbeitung gezeigt wird. Die in dieser Arbeit vorgestellten Konzepte wurden teilweise aufbauend auf dem WoMIS-Prototypen erstellt. Die Funktionalitäten des Systems werden im Folgenden kurz aufgeführt (Tabelle 16 gibt einen Überblick über die einzelnen Funktionalitäten und den entsprechenden Realisierungsumfang).

In Bezug auf das beschriebene Konzept beinhaltet WoMIS zwei Haupteinschränkungen: Erstens bietet WoMIS im Vergleich zu den in Abbildung 32 dargestellten kontextbildenden Objekttypen lediglich Kontexte zu den kontextbildenden Objekttypen „Prozess“, „Aktivität“ und „Geschäftsfall“, die in einer hierarchischen Liste dem Anwender zur Auswahl dargestellt werden. Zweitens enthält WoMIS ein sehr eingeschränktes Informationsmodell von Kontextdokumenten: Von den in Abschnitt 6.3 dargestellten Eigenschaften sind die Aspekte „Identifikation“, „Inhalt“ und „Adressaten und Berechtigungen“ teilweise umgesetzt. Die Aspekte „Sichtbarkeit“ und „Bereitstellung“ wurden nicht realisiert.

Tabelle 16: Übersicht über die Funktionalitäten von WoMIS

Funktion	Realisierung
Kontextbildende Objekttypen	<ul style="list-style-type: none"> • Vereinfachtes Objektmodell: Kontexte zu „Prozess“, „Geschäftsfall“ und „Aktivität“ werden unterstützt.
Visualisierung von/Navigation auf Kontexten	<ul style="list-style-type: none"> • Darstellung der Kontextmenge als hierarchische Liste (Baumstruktur) • Vereinfachtes Konzept des aktuellen Kontexts; dieser besteht aus dem Kontext der aktuell durchzuführenden Aktivität.
Informationsmodell von Kontextdokumenten	<ul style="list-style-type: none"> • Stark vereinfachtes Informationsmodell • Aspekte „Identifikation“, „Inhalt“ und „Adressaten und Berechtigungen“ teilweise umgesetzt • Aspekte „Sichtbarkeit“ und „Bereitstellung“ nicht umgesetzt
Operationen auf Kontextdokumenten	<ul style="list-style-type: none"> • Einfügen, Löschen, Suchen von Kontextdokumenten • Ändern von Attributwerten zu Kontextdokumenten • Systemunterstütztes Einfügen von Kontextdokumenten nicht realisiert
Systemdokumente	<ul style="list-style-type: none"> • Nicht realisiert
Integration in Workflow-Client	<ul style="list-style-type: none"> • Integration in CSE WebDesk • WoMIS wird aus CSE WebDesk heraus im Modus „aktueller Kontext“ geöffnet bzw. aktualisiert.

Die Basisoperationen auf Kontexten und Kontextdokumenten zur Unterstützung der Durchführung wissensintensiver Prozesse wurden in WoMIS umgesetzt. Darüber hinaus wurde WoMIS in die Komponente WebDesk in der Form integriert, dass beim Start von WoMIS aus WebDesk heraus der aktuelle Kontext (hier vereinfacht: der Kontext zur aktuell zu bearbeitenden Aktivität) angezeigt wird.

Darstellung von Kontexten und Kontextdokumenten

Die Visualisierung von Kontexten und möglichen Inhalten in WoMIS zeigt Abbildung 44. Bei WoMIS handelt es sich um ein auf Internet-Technologien basierendes System, so dass ein herkömmlicher WWW-Browser als Client dient. Die links dargestellte Baumstruktur stellt die vorhandenen Kontexte hierarchisch dar. Dazu werden die Informationen über definierte Prozesse, die zugehörigen Aktivitäten und laufende sowie abgeschlossene Geschäftsfälle aus CSE Workflow durch WoMIS ausgelesen und angezeigt (in der Abbildung beispielhaft am Prozess "Contract Management" dargestellt). Der Benutzer kann in dieser Baumstruktur navigieren und einen Kontext auswählen, dessen zugehörige Kontextdokumente daraufhin im rechten Teil des Clients angezeigt werden.

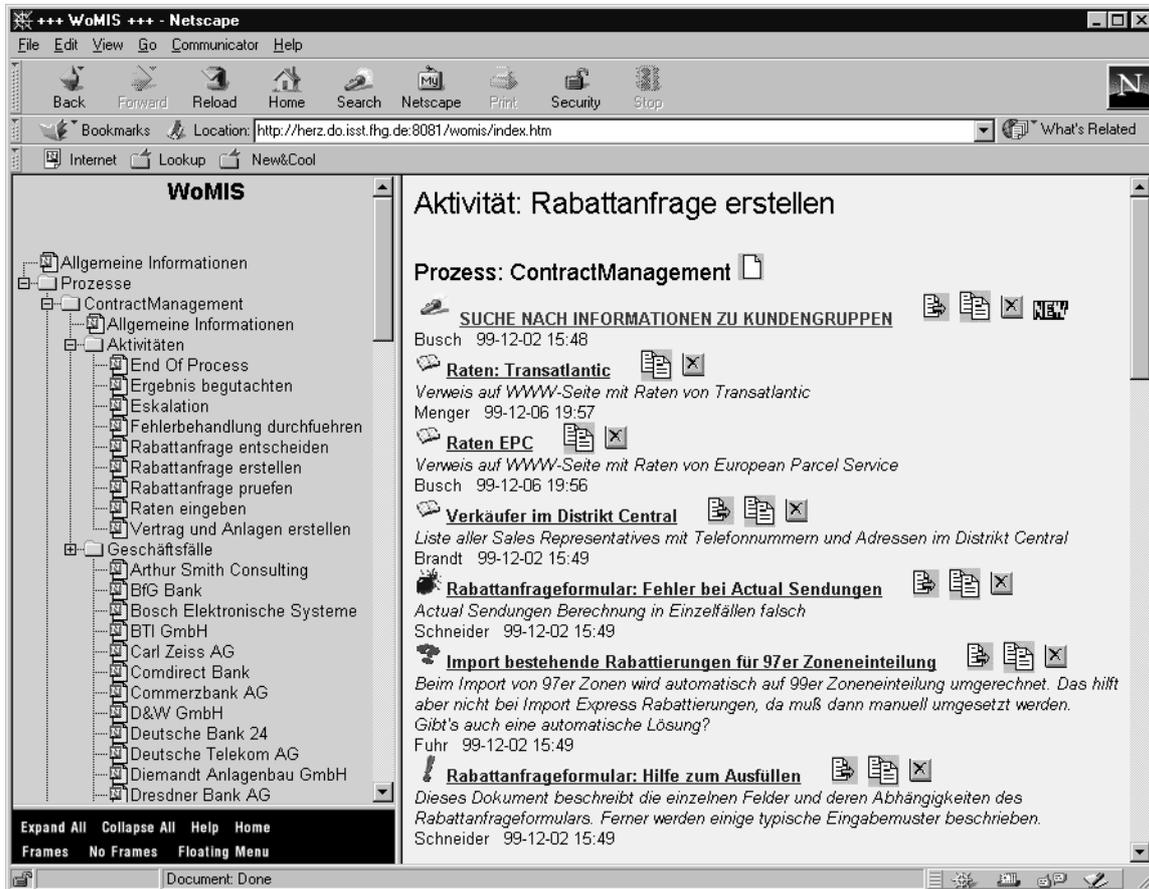


Abbildung 44: Darstellung von Kontexten und Kontextdokumenten in WoMIS

Die Typen der einzelnen Kontextdokumente werden durch unterschiedliche Icons verdeutlicht. Zusätzlich wird zu jedem Kontextdokument der Titel (dient als Hyperlink im Fall eines dateibasierten Kontextdokuments), eine Beschreibung, der Autor und der Zeitpunkt der letzten Änderung angezeigt. Die Operationen zum Einfügen eines Kontextdokuments, Löschen eines Kontextdokuments sowie zum Anzeigen bzw. Ändern der Attributwerte werden ebenfalls durch Icons dargestellt.

Abbildung 45 zeigt die Maske zur Modifikation von Attributwerten von Kontextdokumenten. Die folgenden Attribute wurden in WoMIS realisiert:

- Autor
- Titel
- Beschreibung
- Typ
- Berechtigung „Eingeschränkt sichtbar“
- Identifikation

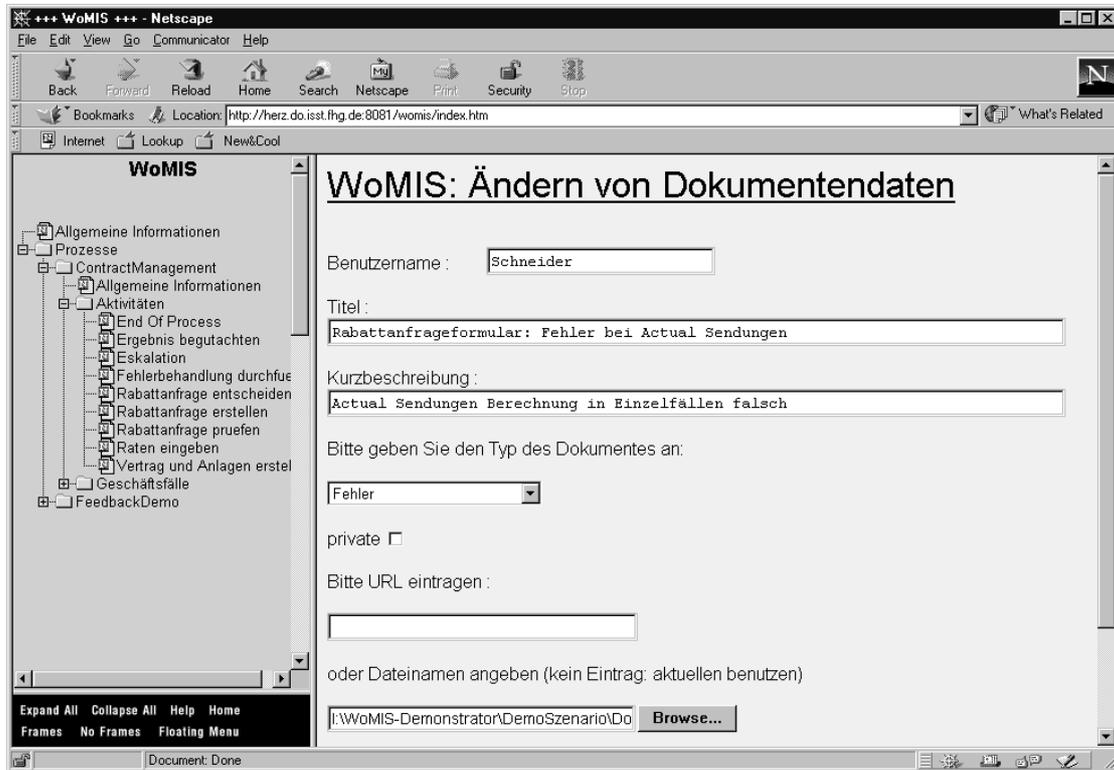


Abbildung 45: Ändern von Attributwerten von Kontextdokumenten in WoMIS

Integration von WoMIS in die Bearbeitung von Geschäftsfällen

Bei der Durchführung von Geschäftsfällen gilt folgende Aufgabentrennung zwischen WebDesk und WoMIS: Die Anzeige der Arbeitsliste, die Annahme der Bearbeitung eines bestimmten Geschäftsfalls, die Bearbeitung der Prozessmodelldokumente, etc. wird weiterhin ausschließlich mittels WebDesk durchgeführt, während die Verwaltung der sekundären Dokumente (Kontextdokumente) durch WoMIS erfolgt.

Um nun trotz dieser Trennung einen in die Geschäftsfallbearbeitung integrierten Zugriff auf die durch WoMIS verwalteten Kontexte zu ermöglichen, wurde die Komponente WebDesk geringfügig modifiziert. Dazu zeigt Abbildung 46 die Arbeitsliste eines Benutzers in der modifizierten Komponente WebDesk: Zu jedem Geschäftsfall kann einerseits (in jeder Zeile rechts durch das Dokumenten-Icon dargestellt) der Zugriff auf die elektronische Umlaufmappe und damit auf die Prozessmodelldokumente erfolgen, die weiterhin durch das WFMS verwaltet werden. Andererseits enthält der modifizierte WebDesk für jeden Geschäftsfall einen Link zum WoMIS-System, der bei Aktivierung in einem separaten Fenster den WoMIS-Client öffnet (wie in Abbildung 44 dargestellt). WoMIS zeigt daraufhin als aktuellen Kontext den Kontext der in diesem Geschäftsfall aktuell zu bearbeitenden Aktivität an. Befindet sich also beispielsweise der Geschäftsfall "Bosch Elektronische Systeme" zur Zeit in der Aktivität "Rabattanfrage erstellen", visualisiert WoMIS den Kontext zu dieser Aktivität.

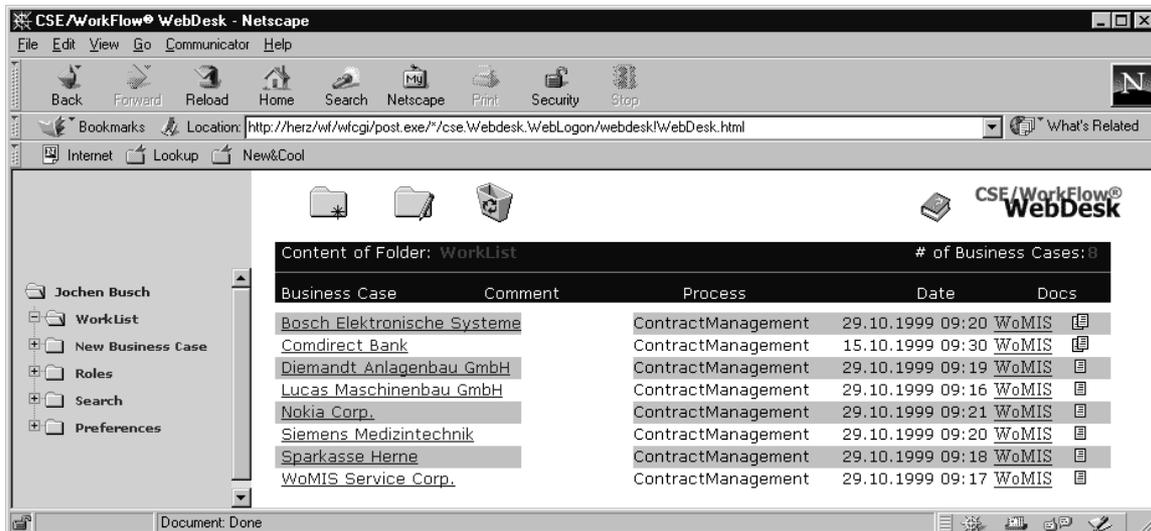


Abbildung 46: Integration von WoMIS in CSE WebDesk

Für das Beispiel "Contract Management" könnte eine typische Interaktion eines Verkäufers mit dem WFMS und WoMIS z.B. in folgenden Schritten ablaufen:

- Der Verkäufer sieht in der elektronischen Umlaufmappe von WebDesk, dass für den Geschäftsfall "Bosch elektronische Systeme" eine Rabattanfrage erstellt werden muss.
- Dazu öffnet er in WebDesk die elektronische Umlaufmappe, die die entsprechende Rabattanfrage enthält. Aufgrund der schwierigen Verhandlungen mit dem Kunden ist sich der Verkäufer unsicher, ob er dem Kunden die geforderten Konditionen anbieten darf. Es sind also weitere Informationen notwendig.
- Der Verkäufer startet aus WebDesk heraus WoMIS, worauf WoMIS den Kontext zur aktuellen Aktivität "Rabattanfrage erstellen" (aktueller Kontext) anzeigt.
- Der Verkäufer öffnet in WoMIS ein Dokument, das seinen Handlungsspielraum bei der Rabattierung beschreibt. Zusätzlich greift er auf die Informationen zu Wettbewerbern zu.
- Mit Hilfe dieser Informationen füllt der Verkäufer die Rabattanfrage aus und nutzt den vorgegebenen Spielraum bei der Rabattierung optimal aus. Daraufhin sendet er den Geschäftsfall in WebDesk an den nächsten Bearbeiter weiter.

9.2.2 Architektur und Implementierung

Abschließend soll die softwaretechnische Realisierung von WoMIS skizziert werden. Dazu zeigt Abbildung 47 die Architektur von WoMIS integriert in CSE Workflow. Die WoMIS-Architektur entspricht grundsätzlich dem im vorherigen Kapitel diskutierten Architekturvorschlag für erweiterte WFMS - umgesetzt auf Internet-Technologien.

Der WoMIS-Client besteht aus einer Menge von HTML-Seiten, die vom WWW-Server zur Laufzeit dynamisch erzeugt werden und daraufhin vom WWW-Browser auf dem Client angezeigt werden. Die Server-Komponente von WoMIS wurde durch eine Menge von ausführbaren Programmen realisiert, die jeweils einzelne Funktionalitäten, wie z.B. zur Anzeige der Dokumentenliste, zur Modifikation von Attributwerten zu einem Kontextdokument oder zum Löschen eines Kontextdokuments, implementieren und über das Common Gateway Interface (CGI) des WWW-Servers aufgerufen werden. Ein separates Datenbank-Modul kapselt den Zugriff auf sowohl die WoMIS-Datenbank als auch die Workflow-Datenbank. Hier weicht die WoMIS-Architektur in geringem Maße von dem in Kapitel 8 vorgestellten Architekturvorschlag ab, da aufgrund fehlender Funktionen des Workflow-Servers zum Lesen bestimmter Inhalte der Workflow-Datenbank der WoMIS-Server direkt lesend auf die Workflow-Datenbank zugreift.

Wie im vorherigen Kapitel gefordert, musste der Workflow-Server nicht modifiziert werden. Es waren lediglich geringe Änderungen der HTML-Seiten des WebDesk und der NoLimits-Klassenbibliothek notwendig, um die oben beschriebene Integration von WebDesk und WoMIS durchführen zu können.

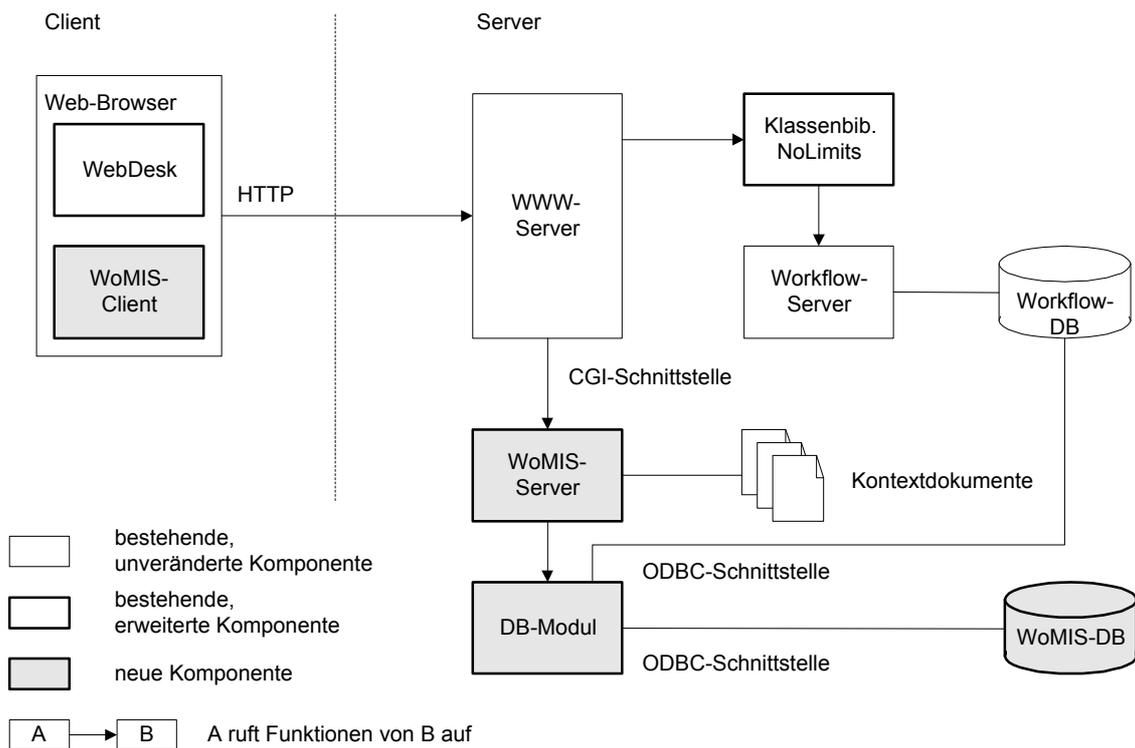


Abbildung 47: Architektur von WoMIS

Die WoMIS-Datenbank wurde als MS-Access-Datenbank realisiert. Sowohl das Datenbank-Modul als auch die CGI-Programme des WoMIS-Servers wurden in der Programmiersprache Perl implementiert. Dabei erfolgte der Zugriff auf die Datenbanken über die ODBC-Schnitt-

stelle. Als WWW-Server wurde der MS Internet Information Server eingesetzt. Zur Visualisierung der Kontexte in der Navigationssicht wurde die Freeware-Komponente Joust [Jous] genutzt.

9.3 Das prototypische System KontextNavigator

Wie im vorherigen Abschnitt beschrieben wurde, handelt es sich bei WoMIS um einen frühen Prototypen, der lediglich die Basiskonzepte von Kontexten realisiert. Um das in Kapitel 6 beschriebene Konzept möglichst weit prototypisch umsetzen zu können, ergab sich die Notwendigkeit einer Neuentwicklung. Das entsprechende System KontextNavigator wird in diesem Abschnitt beschrieben (siehe auch [Goes01a, Goes01b]; weitere Erläuterungen insbesondere zur technischen Umsetzung finden sich in [Liss01]). Der folgende Abschnitt 9.3.1 gibt zunächst einen Überblick über die implementierten Funktionalitäten, die daraufhin in Abschnitt 9.3.2 insbesondere in Bezug auf die entsprechende Benutzerinteraktion detaillierter vorgestellt werden. Abschnitt 9.3.3 stellt die Software-Architektur des Systems vor und erläutert die Funktion der einzelnen Komponenten. Der Aufbau der Kontextdatenbank wird in Abschnitt 9.3.4 vorgestellt. Schließlich werden in Abschnitt 9.3.5 spezifische Implementierungsaspekte, wie z.B. die verwendeten Softwareprodukte, aufgeführt.

9.3.1 Funktionalität

KontextNavigator realisiert die zentralen Aspekte des in dieser Arbeit beschriebenen Konzepts und geht damit über die in WoMIS implementierten Funktionalitäten hinaus. Tabelle 17 gibt einen Überblick über die Funktionalitäten von KontextNavigator.

In Bezug auf die kontextbildenden Objekttypen wurde das Konzept fast vollständig umgesetzt (dabei ergaben sich aus dem Metamodell von CSE Workflow heraus nur geringe Einschränkungen). Im Vergleich zu WoMIS wurden insbesondere Kontexte zu Objekten der Organisationsstruktur (Bearbeiter, Rollen, hierarchische Organisationseinheiten) realisiert. Nicht implementiert wurde der kontextbildende Objekttyp „Aktivitätsblock“. KontextNavigator visualisiert die Kontextmenge, wie in Abschnitt 8.2.2 beschrieben und in Abbildung 40 skizziert, als graphische Struktur. KontextNavigator bietet dazu die beiden Modi „Freie Navigation“ und „Aktueller Kontext“ an. Das Konzept des aktuellen Kontexts wurde analog zur Abbildung 41 als Zusammenfassung mehrerer Kontexte umgesetzt.

In Bezug auf das Informationsmodell von Kontextdokumenten erfolgte eine Einschränkung im Aspekt „Bereitstellung“. KontextNavigator erlaubt die Spezifikation von aktiv bereitzustellenden Dokumenten, die dem Bearbeiter automatisch in einer gesonderten Liste zu dem Zeitpunkt angezeigt werden, wenn der Bearbeiter einen bestimmten Kontext öffnet. Damit realisiert KontextNavigator nicht die Reaktion auf weitere Ereignisse, wie in Tabelle 13 aufgeführt. Die weiteren Aspekte des Informationsmodells wurden weitgehend implementiert. Insbesondere erlaubt KontextNavigator die Angabe von Sichtbarkeitsbedingungen, die es erlauben, dem Bearbeiter bestimmte Kontextdokumente in Abhängigkeit von Variablenwerten des aktuell bearbeiteten Geschäftsfalls anzuzeigen. Die Operationen auf Kontextdokumenten wurden ähnlich wie bei dem Prototypen WoMIS umgesetzt. Ein systemunterstütztes Einfügen von

Kontextdokumenten wurde nicht realisiert. Von KontextNavigator verwaltete Kontexte können Systemdokumente enthalten. Es wurden einige Typen von Systemdokumenten realisiert, die jeweils Abfragen unterschiedlicher Komplexität auf dem WFMS-Datenbestand beinhalten und die Ergebnisse als MS-Word-Dokumente visualisieren. Schließlich wurde KontextNavigator in ähnlicher Form wie WoMIS in den CSE WebDesk integriert.

Tabelle 17: Übersicht über die Funktionalitäten von KontextNavigator

Funktion	Realisierung
Kontextbildende Objekttypen	<ul style="list-style-type: none"> • Kontextbildende Objekttypen, soweit von CSE Workflow unterstützt, fast vollständig realisiert, insbesondere die kontextbildenden Objekttypen der Organisationsstruktur • Kontextbildender Objekttyp „Aktivitätsblock“ nicht umgesetzt
Visualisierung von/Navigation auf Kontexten	<ul style="list-style-type: none"> • Visualisierung der Kontextmenge als graphische Struktur (wie in Abbildung 40 dargestellt) • Realisierung der beiden Modi „Freie Navigation“ und „Aktueller Kontext“
Informationsmodell von Kontextdokumenten	<ul style="list-style-type: none"> • Informationsmodell für Kontextdokumente größtenteils umgesetzt • Kernelemente der Aspekte „Identifikation“, „Inhalt“, „Lebenszyklus“, „Adressaten und Berechtigungen“ und insbesondere auch „Sichtbarkeit“ realisiert • Aspekt „Bereitstellung“ vereinfacht realisiert
Operationen auf Kontextdokumenten	<ul style="list-style-type: none"> • Einfügen, Löschen, Suchen von Kontextdokumenten • Ändern von Attributwerten von Kontextdokumenten • Systemunterstütztes Einfügen von Kontextdokumenten nicht realisiert
Systemdokumente	<ul style="list-style-type: none"> • Verschiedene Typen von Systemdokumenten realisiert
Integration in Workflow-Client	<ul style="list-style-type: none"> • Integration in CSE WebDesk • KontextNavigator kann aus CSE WebDesk heraus über verschiedene Aktivierungsmöglichkeiten in den beiden Modi „Freie Navigation“ und „Aktueller Kontext“ aufgerufen werden.

9.3.2 Benutzerinteraktion

Die Funktionalität von KontextNavigator wird in diesem Abschnitt detaillierter erläutert. Dazu werden die entsprechenden Bildschirmmasken dargestellt.

9.3.2.1 Überblick über die Benutzerschnittstelle

Abbildung 48 zeigt die Visualisierung von Kontexten und Kontextdokumenten in KontextNavigator. Analog zur abstrakten Darstellung des Kontext-Clients im vorherigen Kapitel (Abbildung 40) gliedert sich die Benutzerschnittstelle in einen Navigationsbereich, ein Funktionsmenü und eine Dokumentenliste.

Der Navigationsbereich visualisiert die Kontextstruktur graphisch. Im dargestellten Beispiel ist als zentral dargestellter aktiver Kontext der Kontext zum Prozess „Contract Management“ ausgewählt. Der Navigationsbereich zeigt weiterhin die in der Kontextstruktur übergeordneten bzw. untergeordneten Kontexte sowie Kontexte auf der gleichen Ebene wie der aktive Kontext (siehe dazu auch den folgenden Abschnitt). Durch die Auswahl mit der Maus kann der Benutzer einen neuen aktiven Kontext wählen, worauf sich die Darstellung des Navigationsbereichs automatisch anpasst, indem der neue aktive Kontext in die zentrale Position des Navigationsbereichs bewegt wird. Somit kann der Benutzer auf einfache Weise über die gesamte Kontextstruktur navigieren.

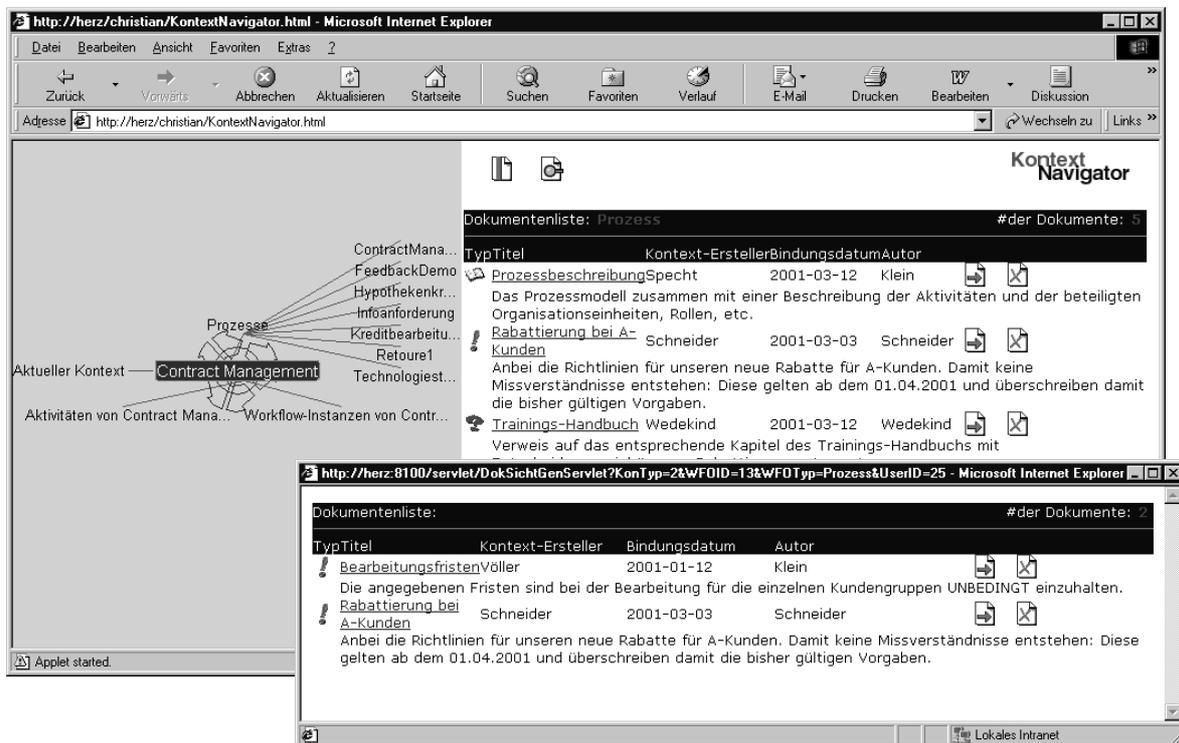


Abbildung 48: Darstellung von Kontexten und Kontextdokumenten in KontextNavigator

Die Dokumentenliste zeigt stets die Kontextdokumente des aktiven Kontexts an (und wird somit ebenfalls aktualisiert, falls der Benutzer einen neuen aktiven Kontext auswählt). Durch Anklicken des Verweises in der Spalte „Titel“ kann der Benutzer ein Dokument öffnen. In einem separaten Fenster, das automatisch geöffnet wird, zeigt KontextNavigator die Dokumente an, die aktiv durch das System bereitgestellt werden sollen, falls der Benutzer in den

entsprechenden Kontext navigiert. Über Schaltflächen kann der Benutzer sich die Attributwerte zu einem Kontextdokument anzeigen lassen und diese ändern sowie ein Dokument aus dem Kontext entfernen.

Über das Funktionsmenü kann der Benutzer die folgenden Operationen ausführen:

- Dokument in den aktiven Kontext einfügen
- Suchen von Dokumenten über alle Kontexte anhand von Attributwerten

Wie in Abschnitt 8.2.2 dargestellt, soll der Kontext-Client zwei Modi unterstützen, nämlich die freie Navigation innerhalb der Kontextstruktur und den Zugriff auf den aktuellen Kontext. Abbildung 49 zeigt diesbezüglich die Darstellung des aktuellen Kontexts in KontextNavigator. Der aktuelle Kontext setzt sich zusammen aus den folgenden drei Kontexten:

- Kontext des Prozesses, dem der aktuelle Geschäftsfall zugeordnet ist (hier: „Contract Management“)
- Kontext des Geschäftsfalls, den der Bearbeiter aktuell durchführt (hier: „BFG Bank“)
- Kontext der Aktivität (im aktuellen Geschäftsfall), die der Bearbeiter gerade durchführt (hier: „Raten eingeben“)

Die Kontextdokumente dieser drei Kontexte werden in der Dokumentensicht von KontextNavigator untereinander dargestellt, um somit dem Bearbeiter einen direkten Zugriff auf die Dokumente größter Relevanz zu ermöglichen. In der Navigationssicht wird der aktuelle Kontext als aktiver Kontext dargestellt.

Die Integration zwischen CSE WebDesk und KontextNavigator wurde ähnlich wie beim Prototypen WoMIS durchgeführt. Dabei wurde CSE WebDesk folgendermaßen erweitert: Über einen neuen Eintrag „KontextNavigator“ in der in Abbildung 46 links dargestellten Navigationsstruktur von CSE WebDesk lässt sich KontextNavigator im Modus „Freie Navigation“ starten. Aktiviert der Benutzer dagegen einen Verweis, der in der rechts dargestellten Arbeitsliste neben einem Geschäftsfall dargestellt ist, startet KontextNavigator im Modus „aktueller Kontext“.

9.3.2.2 Struktur und Visualisierung der Kontexte im Navigationsbereich

Abbildung 50 zeigt die realisierte Navigationsstruktur über die Menge der Kontexte in KontextNavigator. Dabei stellen die grau hinterlegten Objekte abstrakte Kontexte dar, d.h. Kontexte, die lediglich zur Strukturierung der Kontextmenge dienen und die keine Dokumente enthalten können. Die dargestellten Kontexte und ihre Beziehungen sind bereits ausführlich erläutert worden. Es sei an dieser Stelle lediglich auf zwei Aspekte hingewiesen: KontextNavigator erlaubt erstens eine hierarchische Repräsentation der Organisationseinheiten und visualisiert zweitens Verweise, die über eine hierarchische Baumstruktur hinausgehen. Diese Verweise sind entweder statisch, d.h. sie werden bei der Erzeugung der Kontextstruktur fest erzeugt, oder dynamisch, d.h. in Abhängigkeit von der Bearbeitungssituation werden neue Verweise zwischen Kontexten festgelegt.

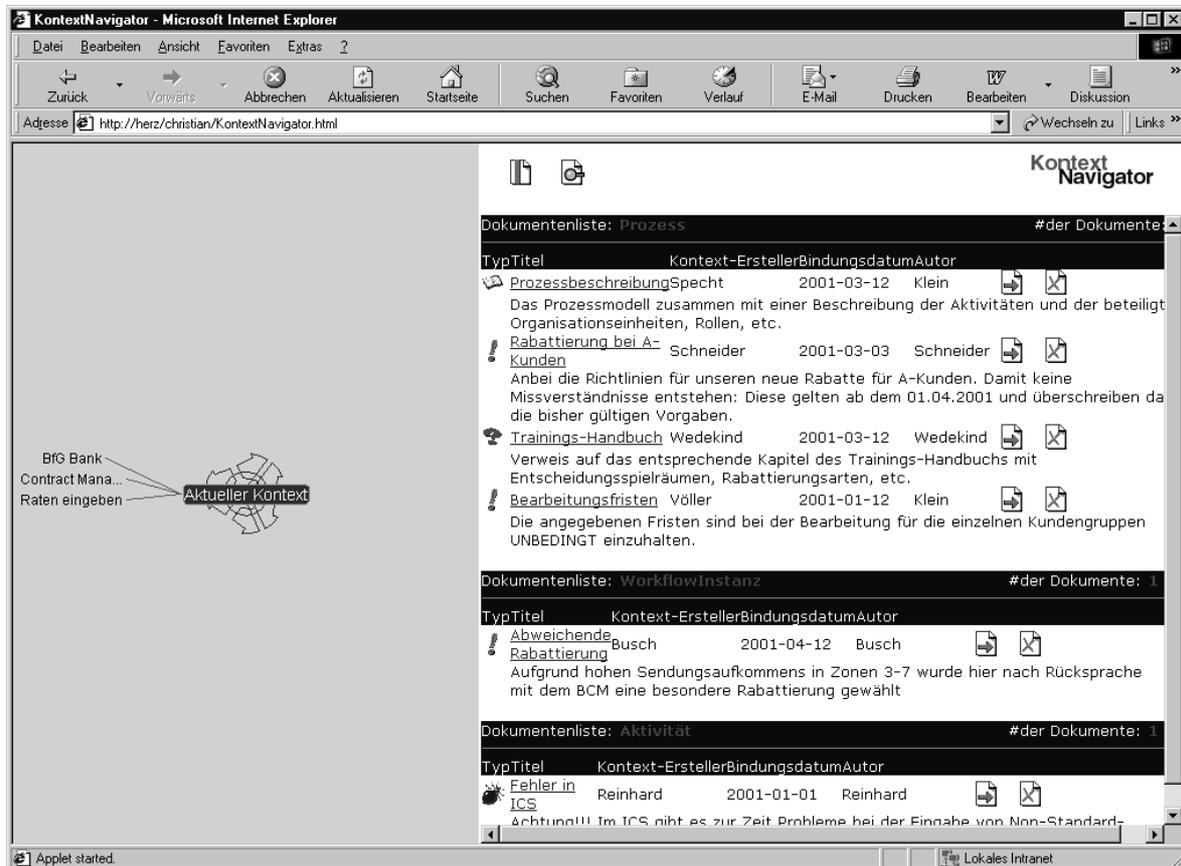


Abbildung 49: Darstellung des aktuellen Kontexts in KontextNavigator

Statische Verweise bestehen von einem Prozess aus zu den Aktivitäten, aus denen sich der Prozess zusammensetzt, sowie von einer Aktivität aus zu den Bearbeitern bzw. Rollen, die im Prozessmodell als potentielle Bearbeiter für die Aktivität festgelegt sind. Dynamische Verweise werden erzeugt, wenn der Bearbeiter KontextNavigator im Modus „Aktueller Kontext“ startet. In diesem Fall wird der aktuelle Kontext (wie oben beschrieben) in Abhängigkeit von den Eigenschaften des aktuell zu bearbeitenden Geschäftsfalls erzeugt, und es werden dynamisch Verweise auf die Kontexte gesetzt, aus denen sich der aktuelle Kontext zusammensetzt. Über diese Verweise kann der Benutzer vom aktuellen Kontext aus in die Gesamtstruktur der verfügbaren Kontexte hineinnavigieren (wie in Abbildung 50 dargestellt).

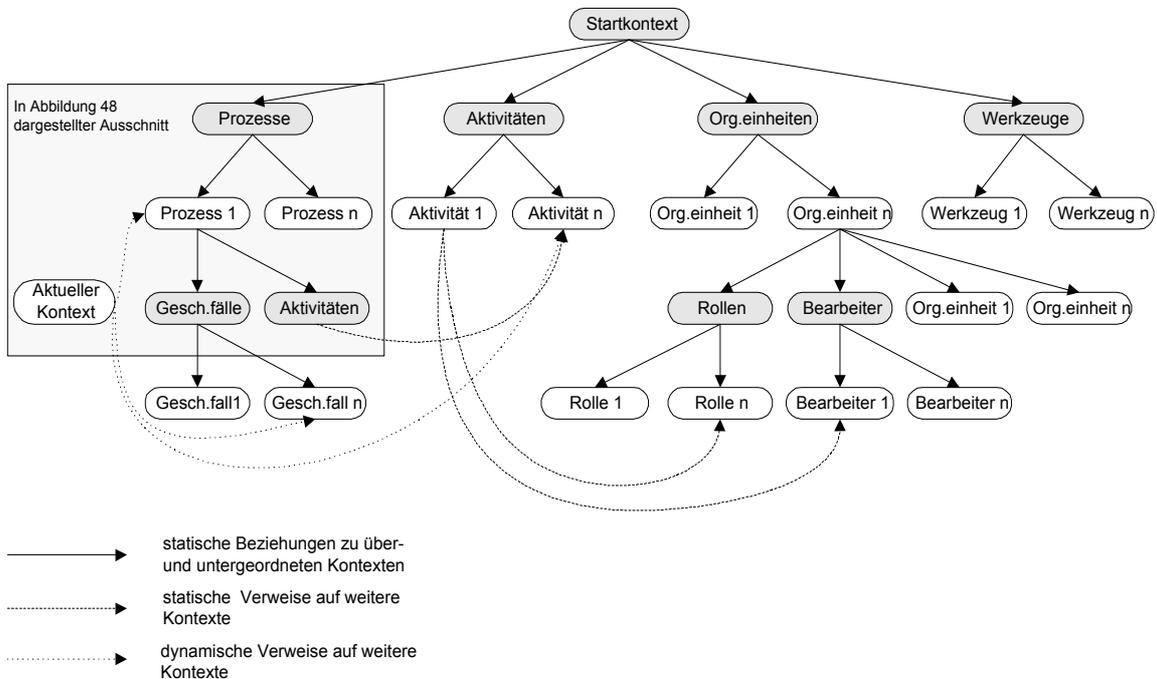


Abbildung 50: Kontextstruktur in ContextNavigator

Die Darstellung von Kontexten im Navigationsbereich von ContextNavigator folgt den folgenden Prinzipien: Zentral wird jeweils der durch den Benutzer ausgewählte, aktive Kontext dargestellt, der zu anderen Kontexten jeweils in einem unterschiedlichen Bezug stehen kann:

- Der übergeordnete Kontext wird oberhalb des aktiven Kontexts dargestellt.
- Untergeordnete Kontexte werden unterhalb des aktiven Kontexts visualisiert.
- Kontexte auf der gleichen Hierarchiestufe wie der aktive Kontext werden rechts neben dem übergeordneten Kontext dargestellt.
- Informationen, die zum aktiven Kontext durch statische oder dynamische Verweise in Beziehung stehen, werden links neben dem aktiven Kontext aufgeführt.

Abbildung 48 und Abbildung 49 zeigen Beispiele für die genannten Typen von Beziehungen zwischen Kontexten.

9.3.2.3 Informationsmodell für Kontextdokumente

In Abbildung 51 wird das Informationsmodell für Kontextdokumente aufgezeigt, das ContextNavigator zur Verfügung stellt. Eine Erläuterung der einzelnen Attribute ist aus den entsprechenden Beschreibungen in Abschnitt 6.3 ersichtlich und soll hier nicht noch einmal gegeben werden.

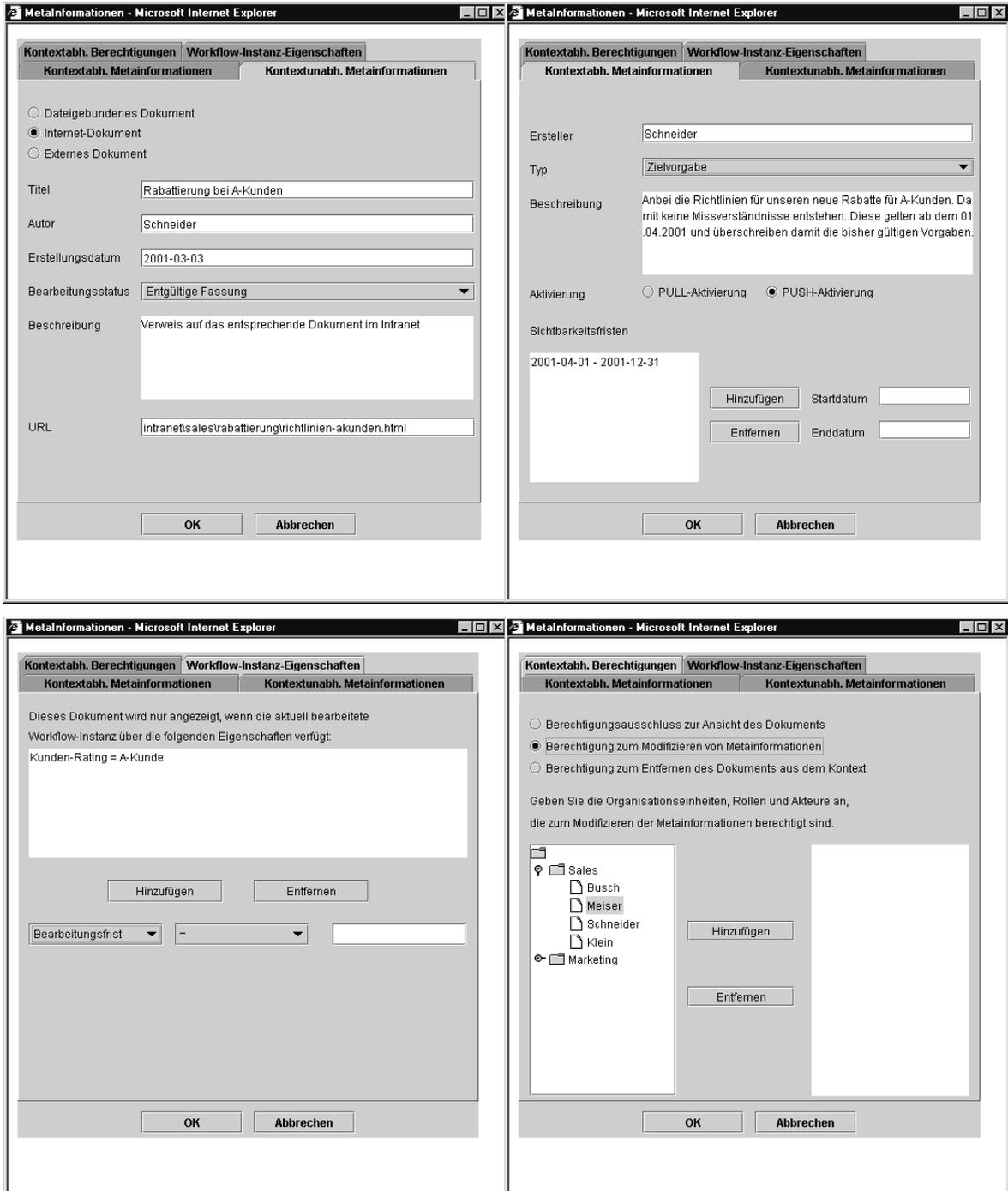


Abbildung 51: Informationsmodell für Kontextdokumente in KontextNavigator

Besondere Beachtung verdient die Registerkarte „Workflow-Instanz Eigenschaften“. Hier ist die Definition von prozessbezogenen Sichtbarkeitsbedingungen möglich. Im angegebenen Beispiel wird das Kontextdokument nur angezeigt, falls der aktuell zu bearbeitende Geschäftsfall (in KontextNavigator mit dem Begriff „Workflow-Instanz“ bezeichnet) die Bedingung „Kunden-Rating = A-Kunde“ erfüllt. Das Dokument, das Richtlinien zur Rabattierung enthält, wird somit nur dargestellt, falls es sich um eine Vertragserstellung für einen Großkunden handelt.

9.3.2.4 Systemdokumente

Um dem Benutzer einen einfachen Zugang zu relevanten Informationen über laufende oder abgeschlossene Prozessbearbeitungen integriert in Kontexte anbieten zu können, wurden einige der in Abschnitt 6.4 aufgeführten Systemdokumente als MS-Word-Dokumente realisiert. Dabei handelt es sich um die folgenden Dokumenttypen:

- Informationen zu einem Geschäftsfall (Komplexitätsstufe 2)
- Informationen zu allen laufenden oder abgeschlossenen Geschäftsfällen eines Prozesses (Komplexitätsstufe 2)
- Häufigkeit der Durchführung einer Aktivität durch die einzelnen Bearbeiter in Geschäftsfällen (Komplexitätsstufe 3)

Systemdokumente können wie andere dateibasierte Kontextdokumente vom Bearbeiter in Kontexten gespeichert und geöffnet werden. Lediglich die Modifikation von Systemdokumenten ist nicht erlaubt, da der Inhalt von Systemdokumenten bei jedem Öffnen dynamisch aus den entsprechenden Inhalten der Workflow-Datenbank erzeugt wird (Abschnitt 9.3.3.3). Abschnitt 52 zeigt beispielhaft ein Systemdokument, das anzeigt, welche Bearbeiter die Aktivität „Rabattanfrage entscheiden“ wie oft und in welchen Geschäftsfällen durchgeführt haben. Dabei werden nur Geschäftsfälle ab einem bestimmten Erstellungsdatum und mit einer gewissen Priorität betrachtet.

9.3.3 Architektur

In diesem Abschnitt stellen wir die softwaretechnische Architektur des Prototypen KontextNavigator vor. Dazu gibt Abschnitt 9.3.3.1 zunächst einen Überblick über die wichtigsten Basismechanismen, auf denen der Prototyp beruht. Abschnitt 9.3.3.2 zeigt die Gesamtarchitektur des Systems eingebettet in CSE Workflow, und Abschnitt 9.3.3.3 stellt die einzelnen Komponenten der Architektur vor.

Informationen zur Durchführung einer Aktivität

Dieses Dokument enthält Informationen zur Durchführung der Aktivität **Rabattanfrage entscheiden** im Process **ContractManagement**. Die folgenden Bearbeiter führten diese Aktivität durch.

Bearbeiter	Anzahl der Aktivitätsdurchführungen
Busch	8
Meiser	5

Die folgende Tabelle zeigt die Geschäftsfälle, in denen die Aktivität durchgeführt wurde

Bearbeiter	Geschäftsfall	Erzeugt am
Busch	Siemens AG	1999-09-29 16:08
	Fraunhofer ISST	1999-09-29 16:09
	Dresdner Bank AG	1999-09-29 16:10
	Commerzbank AG	1999-09-29 16:10
	GF-Data GmbH	1999-09-29 16:11
	Car	
	Deu	
	Nok	
Meiser	Fre:	
	Ma	
	Thy	
	Uni	
	Ehr	

Suchen nach Geschäftsfällen

Datum der Erstellung < 2001-01-01
Priorität < 2

Löschen

Letzten Eintrag Löschen

Kommentar Enthält

OK Zur Liste hinzufügen

Abbildung 52: Systemdokument zur Anzeige der Häufigkeit der Durchführung einer Aktivität

9.3.3.1 Verwendete Basismechanismen

Im Folgenden werden die verwendeten Technologien zur Erstellung von Internet-Anwendungen, JAVA-Applets, JAVA-Servlets und Objektkommunikation durch die JAVA-RMI-API, kurz skizziert. Außerdem wird die Funktion einer externen Komponente zur graphischen Visualisierung der Kontextstruktur im KontextNavigator-Client, das TheBrainSDK, kurz erläutert.

JAVA-Applets

Die objektorientierte Programmiersprache JAVA wurde 1995 von SUN Microsystems vorgestellt und hat sich seither insbesondere durch die Eigenschaft der Plattformunabhängigkeit als Mittel zur Erstellung von Internet-Anwendungen etabliert. JAVA-Programme werden zunächst in Byte Code übersetzt, der zur Runtime von einer Java Virtual Machine (JVM) in das plattformabhängige Zielformat übersetzt und ausgeführt wird.

Während JAVA-Anwendungen von eigenständigen JVM ausgeführt werden, handelt es sich bei JAVA-Applets um Programmcode, der innerhalb eines WWW-Browsers durch dessen JVM ausgeführt wird. Dazu wird der Aufruf und die Parametrierung des Applets in eine HTML-Seite eingebettet. Lädt der WWW-Browser die HTML-Seite von einem WWW-Server, wird der Byte Code des Applets ebenfalls in den WWW-Browser geladen und ausgeführt. Die JAVA-Interpreter der gängigen WWW-Browser erachten den Byte-Code eines JAVA-Applets als unsicher und setzen die folgenden Sicherheitsrestriktionen:

- Ein JAVA-Applet hat keinen Zugriff auf die lokalen Ressourcen und insbesondere das Dateisystem des Client-Systems.
- Ein JAVA-Applet kann ausschließlich eine Netzwerkverbindung zu dem Server aufbauen, von dem es geladen wurde.

In KontextNavigator werden JAVA-Applets zur Darstellung der Struktur der Kontexte und zur Anzeige, Erstellung und Modifikation von Attributwerten von Kontextdokumenten verwendet.

Serverseitige Programmausführung mit JAVA-Servlets

Das JAVA-Servlets-API [Flan99] umfasst eine Menge von Klassen, um die Funktionalität von Servern durch die Ausführung von JAVA-Programmen zu erweitern. Der wesentliche Einsatzzweck besteht dabei in der Erzeugung von dynamischen HTML-Seiten. Im Gegensatz zum im Prototypen WoMIS verwendeten Common Gateway Interface wird bei JAVA-Servlets nicht für jeden Request des Clients ein neuer Betriebssystemprozess gestartet. Inzwischen unterstützen etliche WWW-Server JAVA-Servlets durch eine eingebaute JVM oder können durch entsprechende Zusatzprodukte ergänzt werden. Beim Aufruf eines JAVA-Servlets durch den WWW-Browser wird die entsprechende Klasse von der JVM des WWW-Servers geladen und instanziiert. Anschließend reagiert das erzeugte Objekt auf HTTP-GET und HTTP-POST-Anforderungen. Ähnlich wie bei CGI-Programmen erzeugt das JAVA-Servlet dynamisch HTML-Code, der vom WWW-Browser angezeigt wird.

Für den Prototypen KontextNavigator wird ein JAVA-Servlet verwendet, um in Abhängigkeit vom gewählten Kontext die Dokumentenliste zu visualisieren.

Entfernte Methodenaufrufe mit RMI

Über das Remote-Method-Invocation-API (RMI-API) [Flan99] können Java-Objekte Methoden weiterer Objekte, die sich nicht auf dem lokalen Rechner befinden, aufrufen. Ein Client-Objekt greift dabei auf ein RMI-Objekt auf dieselbe Weise zu wie auf ein lokales Objekt (über Objektreferenz.Methodenname(Parameter)). Die Erstellung eines RMI-Objekts umfasst die folgenden Schritte: Zunächst muss für einen RMI-Server eine JAVA-Schnittstelle spezifiziert werden. Diese Schnittstelle umfasst alle Methoden, die das entsprechende Objekt über das Netzwerk zum Aufruf bereitstellt. Anschließend muss ein RMI-Server erstellt werden, der diese Schnittstelle implementiert. Um zur Laufzeit eine Verbindung zu einem Objekt herstellen zu können, muss dieses über einen Verzeichnisdienst (Naming Service) im Netzwerk bekannt gemacht werden. Ein Client-Objekt kann dann über diesen Verzeichnisdienst mit Hilfe der URL eine Referenz auf das RMI-Objekt erhalten, sofern dieses verfügbar ist.

TheBrainSDK

Das TheBrainSDK (Software Development Kit) [Theb00] der Firma TheBrain Technologies Corp. stellt eine Klassenbibliothek zur Visualisierung und Strukturierung von Informationen zur Verfügung. Dazu stellt das SDK eine Menge von JAVA-Schnittstellen bereit, die in die drei Pakete Datenbank, Benutzeroberfläche und Hilfsklassen gruppiert werden.

Das Paket "Datenbank" dient zur Strukturierung und Speicherung von Informationsmengen („Brains“). Eine Informationsmenge wird in Form eines Graphen gespeichert, in dem Informationen („Thoughts“) durch Verweise („Links“) miteinander verbunden sind.

Das Paket "Benutzerschnittstelle" dient zur Visualisierung von Informationsmengen. Die aktive Information wird stets zentral dargestellt. Weitere Informationen können zu der aktiven Information in einem Bezug als übergeordnete Information („Parent Thought“), untergeordnete Information („Child Thought“), Information auf der gleichen Hierarchieebene („Siblings“) oder Information, auf die per Link verwiesen wird („Jump Thought“), stehen. Die Visualisierung dieser Beziehungen erfolgt wie in Abschnitt 9.3.2.2 erläutert und in Abbildung 48 dargestellt.

Die Benutzerschnittstelle kann die Ereignisse, die der Benutzer mit der Maus auslöst, abfragen und durch entsprechend reagieren. Reaktionen auf diese Ereignisse können durch Überschreiben der entsprechenden Klassen beliebig programmiert werden. Als Basisfunktionalität bietet TheBrain die Navigation durch die Informationsstruktur: Wird eine dargestellte Information durch den Benutzer mit der Maus angewählt, dann wird diese Information als neue aktive Information gesetzt und die Darstellung der umgebenden Informationen entsprechend neu gestaltet.

Schließlich enthält das TheBrainSDK durch das Paket "Hilfsklassen" eine Menge von Klassen, mit denen die Darstellung der Benutzerschnittstelle beeinflusst werden kann

9.3.3.2 Überblick über die Gesamtarchitektur

In den folgenden Abschnitten wird die softwaretechnische Realisierung von KontextNavigator erläutert. Dazu zeigt Abbildung 53 die Architektur von KontextNavigator eingebettet in das WFMS CSE Workflow. Die KontextNavigator-Architektur setzt den im vorherigen Kapitel diskutierten Architekturvorschlag (Abbildung 39) mittels Internet-Technologien um.

Sowohl der in Abbildung 39 dargestellte Workflow-Client als auch der Kontext-Client werden mittels HTML und JAVA-Applets realisiert und innerhalb eines WWW-Browsers ausgeführt. Der KontextNavigator-Server umfasst zwei RMI-Server, die jeweils den Datentransfer zwischen dem zugehörigen JAVA-Applet und der KontextNavigator-Datenbank realisieren, ein JAVA-Servlet zur Darstellung der aktuellen Dokumentensicht und eine aktive Komponente (ReplicatorProcess), die den Transfer der relevanten Daten von der Workflow-Datenbank zur KontextNavigator-Datenbank durchführt. Darüber hinaus verwaltet der KontextNavigator-Server die auf dem Server-Rechner gespeicherten Kontextdokumente (sowohl System- als auch Nicht-Systemdokumente). Im folgenden Abschnitt werden die einzelnen Komponenten detaillierter beschrieben.

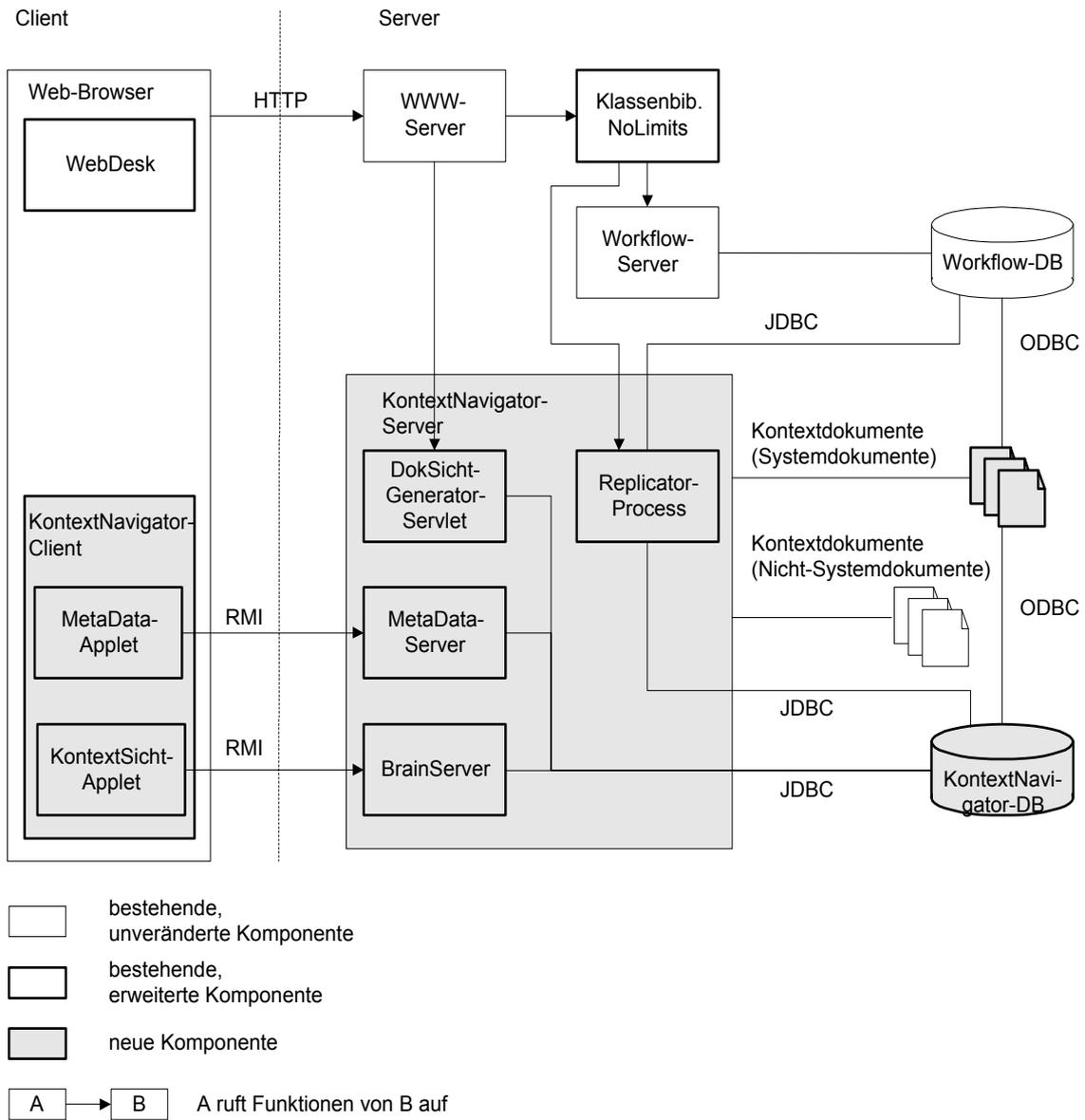


Abbildung 53: Architektur von KontextNavigator

9.3.3.3 Beschreibung der einzelnen Komponenten

In diesem Abschnitt beschreiben wir kurz die Funktion der einzelnen Komponenten und ihre Interaktion. Die Realisierung von KontextNavigator folgt dabei im Wesentlichen der im vorherigen Kapitel vorgestellten Systemarchitektur.

JAVA-Applet „KontextSichtApplet“

Das JAVA-Applet „KontextSichtApplet“ erzeugt die graphische Visualisierung der Kontextstruktur (Abbildung 50) im Navigationsbereich von KontextNavigator (Abbildung 48). Dazu implementiert das Applet eine Schnittstelle des TheBrainSDK, um die Ereignisse, die der Benutzer mit der Maus aktivieren kann, geeignet zu verarbeiten.

Um die entsprechenden Informationen zum Aufbau der Kontextstruktur zu erhalten, ruft das JAVA-Applet bei der Initialisierung den RMI-Server „BrainServer“ auf. Der Darstellungsmodus der Kontextstruktur („Freie Navigation“ oder „Aktueller Kontext“) wird beim Start des JAVA-Applets durch einen Parameter festgelegt und hängt davon ab, ob der Aufruf von KontextNavigator aus dem Standardmenü von CSE WebDesk heraus erfolgt (Modus „Freie Navigation“) oder zur Unterstützung der Bearbeitung eines konkreten Geschäftsfalls dient (Modus „Aktueller Kontext“). Für den Fall, dass der aktuelle Kontext angezeigt werden soll, muss dieser dynamisch in die Kontextstruktur eingehängt werden (Abbildung 49). Nach der Initialisierung des JAVA-Applets und nach jedem Wechsel des aktiven Kontexts erfolgt ein Aufruf des JAVA-Servlets „DokSichtGenerator“, um eine Anpassung der Dokumentenliste von KontextNavigator anzustoßen.

RMI-Server „BrainServer“

Der RMI-Server „BrainServer“ führt den Datentransfer aus der KontextNavigator-Datenbank in das JAVA-Applet „KontextSichtApplet“ durch. Dazu wird eine entsprechende Datenstruktur, die die relevanten Informationen zum Aufbau der Kontextstruktur enthält, erzeugt und an das JAVA-Applet übergeben.

JAVA-Applet „MetaDataApplet“

Das JAVA-Applet „MetaDataApplet“ kann über Schaltflächen des Funktionsmenüs bzw. der Dokumentenliste in unterschiedlichen Modi gestartet werden und führt dementsprechend jeweils die folgenden Aufgaben durch:

- Anzeige von Attributwerten zu einem Kontextdokument
- Modifikation von Attributwerten zu einem Kontextdokument
- Speichern eines neuen Kontextdokuments und der entsprechenden Attributwerte
- Einfügen eines Dokuments in einen weiteren Kontext und Änderung der kontextbezogenen Attributwerte

Zur Durchführung dieser Aufgaben kommuniziert das JAVA-Applet mit dem RMI-Server „MetaDataServer“. Das JAVA-Applet stellt dazu vier Registerkarten bereit (Abbildung 51), in denen je nach gewähltem Modus unterschiedliche Bereiche sichtbar bzw. editierbar sind.

RMI-Server „MetaDataServer“

Der RMI-Server „MetaDataServer“ ist für den Datentransfer zwischen dem JAVA-Applet „MetaDataApplet“ und der KontextNavigator-Datenbank verantwortlich. Darüber hinaus speichert er beim Einfügen eines dateibasierten Kontextdokuments die entsprechende Datei auf dem Server-Rechner.

JAVA-Servlet „DokSichtGeneratorServlet“

Das JAVA-Servlet „DokSichtGeneratorServlet“ erstellt die Dokumentenliste von KontextNavigator und die Liste der in einem separaten Fenster angezeigten Dokumente, die aktiv durch das System bereitgestellt werden sollen (Abbildung 48). Dazu wird das JAVA-Servlet beim Start mit dem Typ des aktiven Kontexts, der entsprechenden Kontext-ID, der User-ID des aktuellen Bearbeiters und der Geschäftsfall-ID des aktuell durchgeführten Geschäftsfalls parametrisiert, falls sich KontextNavigator im Modus „Aktueller Kontext“ befindet.

Handelt es sich beim Typ des aktiven Kontext nicht um einen abstrakten Kontext (Abbildung 49) erzeugt das JAVA-Servlet die zugehörige Dokumentenliste. Dazu werden zunächst aus der KontextNavigator-Datenbank die Informationen über die entsprechenden Kontextdokumente abgefragt. Danach wird für jedes Dokument geprüft,

- ob der Benutzer berechtigt ist, das Dokument angezeigt zu bekommen, und
- ob das aktuelle Datum innerhalb des Gültigkeitszeitraums des Dokuments liegt und
- ob zu dem Dokument eine Sichtbarkeitsbedingung definiert wurde, die in Abhängigkeit von Eigenschaften des aktuell durchgeführten Geschäftsfalls geprüft werden muss.

Die Dokumentenliste und die Liste der aktiv bereitzustellenden Dokumente wird schließlich aus der Menge der Kontextdokumente erzeugt, die die obigen Bedingungen erfüllen.

ReplicatorProcess

Der „ReplicatorProcess“ ist eine aktive Komponente, die bei der Initialisierung des KontextNavigator-Servers die relevanten Attributwerte der kontextbildenden Objekte aus der Workflow-Datenbank ausliest und in die KontextNavigator-Datenbank überträgt. Damit wird die Basis für den Aufbau der Kontextstruktur durch den RMI-Server „BrainServer“ und die Visualisierung der Kontextstruktur durch das JAVA-Applet „KontextSichtApplet“ gelegt. Die Workflow-Datenbank wird von „ReplicatorProcess“ periodisch auf neue Einträge überprüft (z.B. wenn der Workflow-Administrator einen neuen Bearbeiter angelegt hat), die dann in die KontextNavigator-Datenbank eingetragen werden. Darüber hinaus wird die Komponente „ReplicatorProcess“ direkt durch die Klassenbibliothek „No Limits“ informiert, falls der Benutzer in CSE WebDesk einen neuen Geschäftsfall angelegt hat. In diesem Fall wird ein entsprechender Eintrag in der KontextNavigator-Datenbank angelegt. Sonstige Änderungen der Workflow-Datenbank, wie z.B. Löschen von Prozessen oder Aktivitäten, führen nicht zu einem automatischen Abgleich der KontextNavigator-Datenbank, sondern müssen primär über organisatorische Maßnahmen behandelt werden (Kapitel 7).

Systemdokumente

Systemdokumente wurden als MS-Word-Dokumente realisiert. Die Visualisierung der aktuellen Informationen aus der Workflow-Datenbank wurde durch das Erstellen einer neuen Word-Dokumentvorlage erreicht, auf der alle Systemdokumente beruhen. In dieser Dokumentvorlage wurde das AutoOpen-Makro, das bei jedem Öffnen eines zugehörigen Dokuments ausgeführt wird, mit Visual-Basic-Programmcode versehen, der in Abhängigkeit vom Dokumenttyp die entsprechenden Informationen aus der Workflow-Datenbank abfragt und im Dokument visualisiert. Im Einzelnen werden die folgenden Schritte durchgeführt

- Beim Erzeugen eines Systemdokuments werden der Dokumenttyp und die entsprechenden Parameter zur Charakterisierung des Systemdokuments in Word-Dokumentvariablen persistent gespeichert.
- Öffnet der Benutzer das Systemdokument, wird das AutoOpen-Makro ausgeführt.
- Das Makro prüft anhand der in den Dokumentvariablen gespeicherten Parameter, um welchen Dokumenttyp es sich bei dem Systemdokument handelt.
- Falls es sich um einen Dokumenttyp handelt, der Informationen zu einer Menge von Geschäftsfällen darstellt bzw. auswertet, wird dem Benutzer eine Eingabemaske präsentiert, über die er die Menge der Geschäftsfälle geeignet einschränken kann (Abbildung 52).
- Es wird eine Verbindung zur Workflow-Datenbank aufgebaut, und es werden in Abhängigkeit vom Dokumenttyp die entsprechenden Informationen aus der Workflow-Datenbank abgefragt.
- Die Ergebnisse werden in tabellarischer Form im Systemdokument dargestellt.

Systemdokumente können wie andere dateibasierte Dokumente in Kontexten gespeichert werden. Eine Modifikation von Systemdokumenten durch den Bearbeiter ist ausgeschlossen, da der Inhalt von Systemdokumenten bei jedem Öffnen dynamisch erzeugt wird. Das Erzeugen eines Systemdokuments aus dem Kontext-Client heraus durch eine eingebettete Komponente, die eine Standardbibliothek realisiert, wurde nicht implementiert.

9.3.4 KontextNavigator-Datenbank

Abbildung 54 zeigt den Aufbau der relationalen Datenbank „KontextNavigator-DB“. Die Primärschlüssel der Tabellen sind unterstrichen dargestellt. Fremdschlüssel sind in breiter Schrift und nicht unterstrichen aufgeführt. Die Verweise, die Fremdschlüsselabhängigkeiten repräsentieren, zeigen in Richtung der Primärschlüssel.

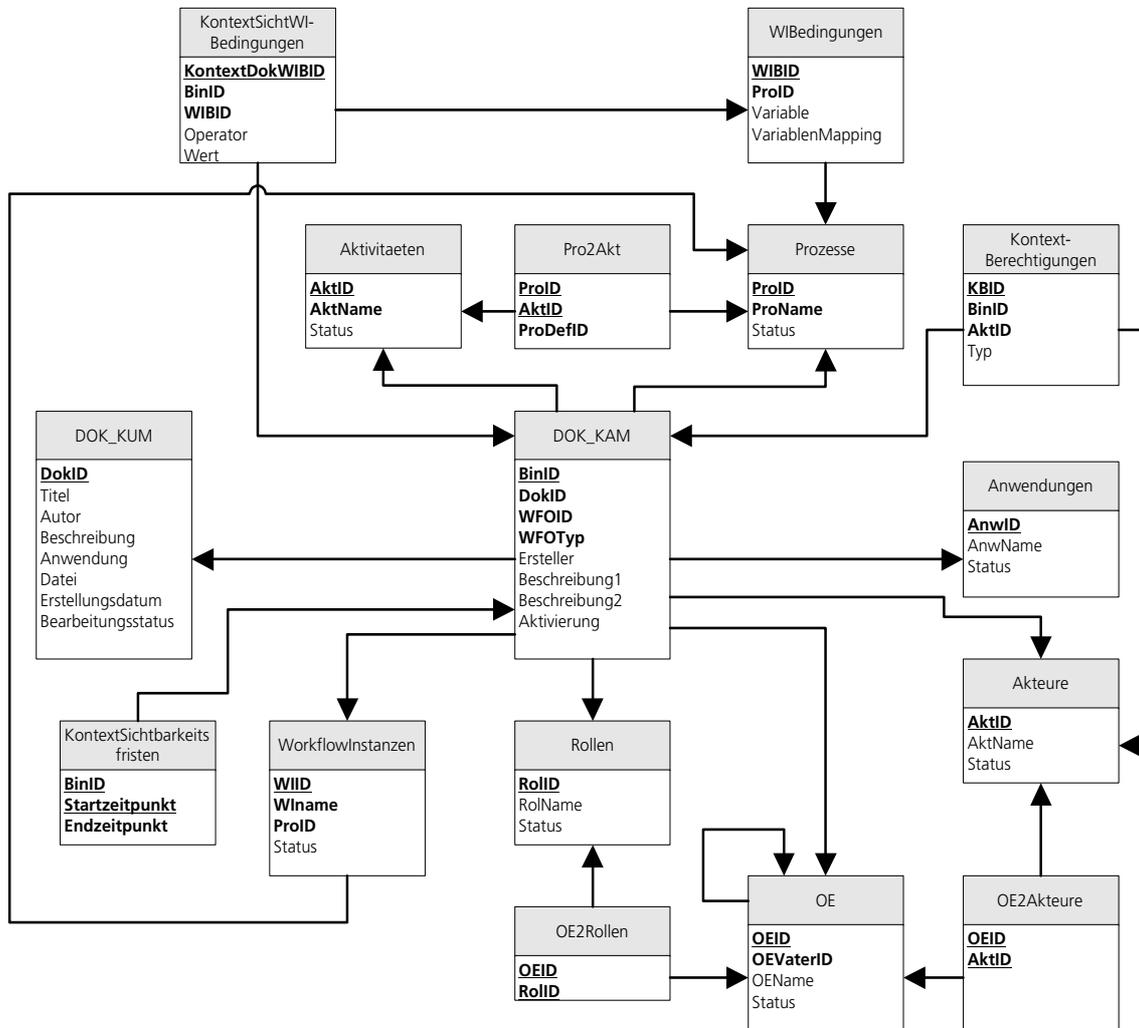


Abbildung 54: Aufbau der Kontextdatenbank von KontextNavigator

Der Aufbau der Datenbank entspricht im Wesentlichen der in Abbildung 42 dargestellten Vorgabe zur Strukturierung einer Kontextdatenbank. Abweichungen ergeben sich einerseits aus der Tatsache, dass Abbildung 54 die Menge der Datenbanktabellen vollständig darstellt (also auch die in Abbildung 42 nicht aufgeführten Tabellen zur Realisierung von many-to-many-Verknüpfungen umfasst), und andererseits aus den in Abschnitt 9.3.1 diskutierten Einschränkungen der Funktionalität von KontextNavigator. Tabelle 18 beschreibt die einzelnen Datenbanktabellen.

Tabelle 18: Beschreibung der Datenbanktabellen

Tabellenname	Beschreibung
Akteure	Beschreibung eines Bearbeiters in CSE Workflow durch seine ID, seinen Namen und seinen Status (aktiv, inaktiv)
Aktivitaeten	Aktivitäten in CSE Workflow charakterisiert durch ihre ID, ihren Namen und ihren Status (aktiv, inaktiv)
Anwendungen	Beschreibung aller Methoden in CSE Workflow durch ihre ID, ihren Namen und ihren Status (aktiv, inaktiv); entspricht dem kontextbildenden Objekttyp „Werkzeug“
Dok_KAM	Beschreibung der kontextbezogenen Attribute von Kontextdokumenten
Dok_KUM	Beschreibung der dokumentbezogenen Attribute von Kontextdokumenten
KontextBerechtigungen	Verwaltung der Berechtigungen für die Sichtbarkeit, das Setzen von Attributwerten und das Entfernen von Dokumenten aus Kontexten für einzelne Benutzer, Rollen oder Organisationseinheiten
KontextSichtbarkeitsfristen	Beschreibung aller Zeiträume, in denen die Sichtbarkeit für ein Dokument in einem Kontext ausgeschlossen wird.
KontextSichtWIBedingungen	Sichtbarkeitsbedingungen für ein Kontextdokument, die unter Verwendung von Sichtbarkeitsvariablen (Tabelle WIBedingungen) gebildet werden und zur Laufzeit ausgewertet werden
OE	Verwaltung aller Organisationseinheiten von CSE Workflow und ihres hierarchischen Aufbaus
OE2Akteure	Zuweisung von Akteuren zu Organisationseinheiten
OE2Rollen	Zuweisung von Rollen zu Organisationseinheiten
Pro2Akt	Zuweisung von Aktivitäten zu den Prozessen, in denen sie verwendet werden
Prozesse	Verwaltung aller Prozesse in CSE Workflow
Rollen	Beschreibung aller Rollen in CSE/Workflow
WIBedingungen	Variablen von Geschäftsfällen, die zur Definition von Sichtbarkeitsbedingungen benutzt werden können. Diese Tabelle wird gemäß Abschnitt 7.1.1 und Abschnitt 6.3.5 vom Modellierer bzw. Administrator mit den relevanten systemspezifischen und prozessspezifischen Variablen belegt.
WorkflowInstanzen	Verwaltung aller Workflow-Instanzen (entspricht Geschäftsfällen) von CSE/Workflow

9.3.5 Implementierungsaspekte

Abschließend werden in diesem Abschnitt Implementierungsaspekte dargestellt, die die konkret verwendeten Softwareprodukte betreffen. Diese werden im Folgenden gegliedert nach Server-Komponenten, Client-Komponenten und Datenbankaspekten aufgeführt.

Server

Bei dem Server-Rechner handelt es sich um einen Microsoft Windows NT 4 Server mit installiertem Service Pack 5. Auf diesem Rechner laufen die folgenden Server-Komponenten der beschriebenen Architektur:

- CSE Workflow Server in der Version „Gold Edition“
- Microsoft Internet Information Server 4.0 als WWW-Server für CSE WebDesk und Kontext-Navigator
- Allaire JRUN-Server als ISAPI-Filter (IIS Application Programming Interface) in den WWW-Server integriert. Der Filter leitet alle Anfragen, die an den WWW-Server gerichtet werden, an den JRUN-Server weiter. Handelt es sich um den Aufruf eines JAVA-Servlets, führt der JRUN-Server diese aus und leitet die Antwort des JAVA-Servlets an den WWW-Server weiter.
- RMI-Server, die durch eine JVM des JDK 1.3 ausgeführt werden. Die Registrierung (Naming Service) erfolgt mit Hilfe des Programms rmiregistry.exe.

Client

In Bezug auf den Client sind hauptsächlich Anforderungen an den WWW-Browser zu betrachten, der das JAVA-SWING-API unterstützen muss. Die Aufrufe der JAVA-Applets wurden so gestaltet, dass sie sowohl unter dem Netscape Navigator (Version 4.7 oder höher) und dem Microsoft Internet Explorer (Version 4.5 oder höher) lauffähig sind. Die Aufrufe der JAVA-Applets wurden dazu mit dem Programm HTMLConv.exe der Firma SUN automatisch erstellt. Dieses Programm erweitert die Aufrufe der JAVA-Applets derart, dass über das Internet mit dem JRE (JAVA Runtime Environment) eine JVM geladen wird, die das SWING-API unterstützt.

Außerdem muss zur Nutzung von KontextNavigator die systemweite Sicherheitskonfigurationsdatei derart angepasst werden, dass sie den Dateitransfer vom Client auf den Server (Upload von Dateien) ermöglicht.

Datenbank

Als Datenbank-Management-System für die Datenbank "KontextNavigator-DB" wurde derselbe Microsoft SQL Server 6.5 verwendet, auf dem auch die Datenbank von CSE/Workflow installiert ist.

E Abschluss

10 Zusammenfassung und Ausblick

10 Zusammenfassung und Ausblick

Ein hoher Anteil der in Organisationen durchgeführten Geschäftsprozesse wird in Zukunft wissensintensiv sein. Wissensintensive Prozesse zeichnen sich durch flexible Wissensbedarfe und -ergebnisse aus, sie enthalten Aktivitäten, die hohe Kommunikations- und Kooperationsbedarfe einschließen oder komplexe Entscheidungen beinhalten, und sie erfordern einen geschäftsfallübergreifenden Wissenstransfer. Workflow-Management-Systeme (WFMS), die bisher bereits die Erwartungen von Herstellern und Anwendern nur in geringem Ausmaß erfüllen konnten, können die Anforderungen, die diese neue Prozessklasse an eine informationstechnische Unterstützung stellt, nur sehr eingeschränkt umsetzen. Mit dem in dieser Arbeit entwickelten Konzept der Kontexte steht ein leistungsfähiger Ansatz der Erweiterung von WFMS zur Unterstützung wissensintensiver Prozesse zur Verfügung.

Das Konzept der Kontexte basiert auf der Integration von relevanten Aspekten aus den Bereichen Workflow Management und Wissensmanagement, um das gesamte für die Prozessbearbeitung relevante Wissen kooperativ aufzubauen, zu strukturieren und dem Bearbeiter integriert in die Prozessbearbeitung verfügbar zu machen. Dazu haben wir die Begriffe des primären und sekundären Wissens eingeführt. Während primäres Wissen im Prozessmodell eines WFMS repräsentiert wird, erfolgt die Verwaltung sekundären Wissens, d.h. des Wissens, dessen Relevanz sich erst situativ in einer konkreten Bearbeitungssituation ergibt oder das erst während der Prozessbearbeitung bekannt ist, durch Kontexte. Beispiele für sekundäres Wissen sind Vorgaben bzgl. der Bearbeitung, Erfahrungen, Gründe für Entscheidungen oder Wissen über ähnliche Geschäftsfälle. Die Strukturierung dieses sekundären Wissens durch Kontexte orientiert sich dabei an den Workflow-Objekten, die dem Bearbeiter bekannt sind, wie z.B. Prozesse (Wissen über den gesamten Prozess), Aktivitäten (Wissen zur Bearbeitung einer bestimmten Aktivität), oder Rollen (Wissen, das die Funktion einer Rolle im Prozess beschreibt). Kontexte enthalten somit jeweils Wissen, das dazu dient, den Bearbeiter in einer konkreten Bearbeitungssituation bei der workflow-gestützten Durchführung von Prozessen zu unterstützen.

Aus einer Analyse bestehender Metamodelle des Workflow Managements wurde eine Menge von Kontexttypen identifiziert, die grundsätzlich für eine Erweiterung unterschiedlichster WFMS geeignet ist. Die Informationen zum Aufbau der Kontextstruktur können somit direkt vom WFMS bereitgestellt werden. Ferner wurde in der Arbeit ein Informationsmodell für Kontextdokumente, d.h. Dokumente, die jeweils das in Kontexten gespeicherte Wissen repräsentieren, erarbeitet, das zwei Kernelemente besitzt: Erstens wurde ein für den Benutzer einfach zu handhabendes, aber leistungsfähiges Konzept zur aktiven Bereitstellung von Kontextdokumenten in Abhängigkeit von Ereignissen der Prozessbearbeitung vorgestellt. Zweitens wurde ein Konzept vorgestellt, das die Relevanz und die Sichtbarkeit von sekundärem Wissen in Beziehung zu Eigenschaften des konkret zu bearbeitenden Geschäftsfalls setzt und somit ermöglicht, dass bestimmte Kontextdokumente z.B. nur im Fall von Ausnahmen bei der Prozessbearbeitung oder bei der Durchführung von Geschäftsfällen für bestimmte Kunden angezeigt werden.

In Bezug auf die organisatorische Einbettung des Kontextkonzepts haben wir die Integration von Maßnahmen zum Aufbau, zur Nutzung und zur Evaluierung von Kontexten in den Prozesslebenszyklus zur kontinuierlichen Verbesserung von Workflow-Management-Anwendun-

gen dargestellt. Dabei haben wir die Rollen Dokumentverantwortlicher, Kontextverantwortlicher und Prozessverantwortlicher zusammen mit ihren Aufgaben eingeführt.

Für eine informationstechnische Umsetzung von Kontexten sind WFMS in verschiedener Hinsicht zu erweitern. Wir haben daher zunächst ein abstrakte Systemarchitektur vorgestellt, die mit geringem Aufwand für die überwiegende Anzahl von kommerziellen WFMS angepasst werden kann, und die Kernelemente dieser Architektur und des Konzepts der Kontexte anhand zweier Prototypen evaluiert. Die beiden Prototypen WoMIS (Workflow Memory Information Systems) und KontextNavigator basieren auf dem kommerziellen WFMS CSE Workflow und setzen die Kernelemente des in dieser Arbeit vorgestellten Konzepts um.

Mögliche Erweiterungen des beschriebenen Konzepts bestehen in einer stärkeren Integration von Kontexten mit den Phasen der Erhebung und Modellierung des Prozesslebenszyklus. Es existieren diesbezüglich bereits verschiedene Ansätze zur wissensorientierten Modellierung von Geschäftsprozessen. Hier sind Konzepte zu entwickeln, wie die Identifikation von sekundärem Wissen und die Strukturierung durch Kontexte bereits integriert in diese frühen Phasen des Prozesslebenszyklus auf methodische Weise durchgeführt werden kann.

Eine zweite Erweiterung betrifft eine stärkere Einbeziehung von Wissensprozessen in die Nutzung von Kontexten. Wissensprozesse umfassen Aktivitäten, wie das Erzeugen, Verteilen oder Nutzen von Wissen, und laufen sowohl parallel als auch orthogonal zu klassischen Geschäftsprozessen ab. Klassische einfache Wissensprozesse sind z.B. Freigabeverfahren, wie sie bereits in Dokumenten- oder Content-Management-Systemen realisiert sind. Die methodische Identifikation und Unterstützung von Wissensprozessen ist noch Gegenstand aktueller Forschungsarbeiten. Eine geeignete Integration in das dargestellte Kontextkonzept wäre in weiterführenden Arbeiten zu untersuchen.

F Literatur und Verzeichnisse

Literatur

Abbildungsverzeichnis

Tabellenverzeichnis

Index

Literatur

- [AAAI00] Proceedings of the AAAI Symposion on Bringing Knowledge to Business Processes. Stanford, USA, AAAI Technical Report, 2000, <http://www.aifb.uni-karlsruhe.de/~sst/Research/Events/sss000/>
- [ABHK98] Abecker, A.; Bernardi, A.; Hinkelmann, K.; Sintek, M.: Techniques for Organizational Memory Information Systems. Forschungsbericht D-98-02, Deutsches Forschungszentrum für Künstliche Intelligenz DFKI, 1998.
- [ABS99] Abecker, A.; Bernardi, A.; Sintek, M.: Enterprise Information Infrastructures For Active, Context-Sensitive Knowledge Delivery. In: Proceedings of ECIS'99 - The 7th European Conference on Information Systems, 1999.
- [Acke94a] Ackerman, M. S.: Augmenting the Organizational Memory: A Field Study of Answer Garden. In: Proceedings of the ACM Conference on Computer Supported Cooperative Work '94 (CSCW'94), S. 243-252.
- [Acke94b] Ackerman, M. S.: Definitional and Contextual Issues in Organizational and Group Memories. In: Proceedings of the 27th Hawaii International Conference on System Sciences (HICSS), Vol. 3, 1994, S. 191-200.
- [AD98] Agostini, A.; De Michelis, G.: Simple Workflow Models. Workflow Management: Net-based Concepts, Models, Techniques and Tools. In: Proceedings of the 19th International Conference on Applications and Theory of Petri Nets, 1998.
- [ADK98] Abecker, A.; Decker, S.; Kühn, O.: Organizational Memory. In: Informatik-Spektrum 21(4), 1998, S. 213-214.
- [AH98] Ackerman, M. S.; Halverson, C.: Considering an Organization's Memory. In: Proceedings of the ACM Conference on Computer Supported Cooperative Work '98 (CSCW'98), S. 39-48.
- [Albr93] Albrecht, F.: Strategisches Management der Unternehmensressource Wissen. Frankfurt: Verlag Peter Lang, 1993.
- [Allw98] Allweyer, T.: Modellbasiertes Wissensmanagement. In: IM Information Management & Consulting, Nr. 1, 1998, S. 37-45.
- [AM90] Ackerman, M. S.; Malone, T. W.: Answer Garden: A Tool for Growing Organizational Memory. In: Proceedings of the ACM Conference on Office Information Systems, 1990, S. 31-39.
- [AM95] Ackerman, M. S.; Mandel, E.: Memory in the Small: An Application to Provide Task-Based Organizational Memory for a Scientific Community. In: Proceedings of the 28th Hawaii International Conference on System Sciences (HICSS), Vol 4, 1995, S. 323-332.
- [AM96] Ackerman, M. S.; McDonald, D. W.: Answer Garden 2: Merging Organizational Memory with Collaborative Help. In: Proceedings of the ACM Conference on Computer Supported Cooperative Work '96 (CSCW'96), S. 97-105.

- [AMB01] Abecker, A.; Maus, H.; Bernardi, A.: Software-Unterstützung für das Geschäftsprozessorientierte Wissensmanagement. In: [SSSS01], S. 39-43 und in: [MAHM01].
- [AN95] Aamodt, A.; Nygard, M.: Different roles and mutual dependencies of data, information and knowledge. In: Data & Knowledge Engineering 16, Elsevier, 1995, S. 191-222.
- [AS78] Argyris, C.; Schön, D.: Organizational Learning: A Theory of Action Perspective, Reading: Addison-Wesley, 1978.
- [BDFL96] Böhm, M.; Deiters, W.; Friedrich, F.; Lindert, F.: Workflow Management as Teleservice. In: Computer Networks and ASDN Systems, Vol 28, No.4, 1996, S. 1961-1969.
- [BFIM98] Berners-Lee, T.; Fielding, R.; Irvine, U. C.; Masinter, L.: Uniform Resource Identifiers (URI): Generic Syntax. RFC2396, W3C Technical Report. <http://www.ietf.org/rfc/rfc2396.txt>
- [BK95] Bogia, D. P.; Kaplan, S. M.: Flexibility and Control for Dynamic Workflows in the wOrlds Environment. In: Proceedings of the Conference on Organizational Computing Systems (COOCS'95), Milpitas, CA, 1995, S. 148-159.
- [BOC01] BOC Information Technologies Consulting GmbH. <http://www.boc-eu.de>
- [BN95] Blumenthal, R.; Nutt, G. J.: Supporting Unstructured Workflow Activities in the Bramble ICN System. In: Proceedings of the ACM Conference on Organizational Computing Systems (COOCS'95), 1995, S. 130-137.
- [BP98] Borghoff, U. M.; Pareschi, R.: (Hrsg.): Information Technology for Knowledge Management. Berlin et al.: Springer, 1998.
- [Bußl98] Bußler, C.: Organisationsverwaltung in Workflow-Management-Systemen. Wiesbaden: Deutscher Universitäts-Verlag, 1998.
- [BVÖ99] Bach, V.; Vogler, P.; Österle, H.: Business Knowledge Management. Berlin et al.: Springer, 1999.
- [BWP97] Bullinger H.-J.; Wörner, K.; Prieto, J.: Wissensmanagement heute. Fraunhofer-Institut für Arbeitswirtschaft und Organisation, Stuttgart, 1997.
- [BWPW98] Bullinger H.-J.; Warschat, J.; Prieto, J.; Wörner, K.: Wissensmanagement - Anspruch und Wirklichkeit: Ergebnisse einer Unternehmensstudie in Deutschland. In: IM Information Management & Consulting, 1 (1998), S. 7-23.
- [CCPP96] Casati, F.; Ceri, S.; Pernici, B.; Pozzi, G.: Workflow Evolution. In: Data and Knowledge Engineering, 24 (3), 1996, S. 211-238.
- [CJ98] Carlsen, S.; Jorgensen, H. D.: Emergent Workflow: The AIS Workware Demonstrator. In Proceedings Workshop „Towards Adaptive Workflow Systems“ of the ACM Conference on Computer Supported Cooperative Work (CSCW'98). <http://ccs.mit.edu/klein/cscw98/>
- [CSE99a] CSE Systems GmbH: NoLimits WF 1.3. Klagenfurt: CSE Systems, 1999.

- [CSE99b] CSE Systems GmbH: WebDesk Browser Based Workflow Management. Klagenfurt: CSE Systems, 1999.
- [CSE99c] CSE Systems GmbH: WorkDesk 5.0 Gold Edition - User Guide. Klagenfurt: CSE Systems, 1999.
- [CW97] Conklin, E. J.; Weil, W.: Wicked Problems - Naming the Pain in Organizations. Working paper, 1997, <http://www.gdss.com/wp>
- [DCEN91] Davis, G.; Collins, R.; Eierman, M.; Nance, W.: Conceptual Model for Research on Knowledge Work. Working paper, University of Minnesota, MISRC-WP-91-10, 1991.
- [DCME99] Dublin Core Metadata Element Set. Version 1.1, <http://dublincore.org/documents/1999/07/02/dces>
- [Deit93] Deiters, W.: A View Based Approach to Software Process Management. Dissertation, Technische Universität Berlin, 1993.
- [Ders96] Derszteler, G.: Workflow Management Cycle - Ein Ansatz zur Integration von Modellierung, Ausführung und Bewertung workflowgestützter Geschäftsprozesse. In: Wirtschaftsinformatik, 38 (1996) 6, S. 591-600.
- [DFL96] Deiters, W.; Friedrich, F.; Lindert, F.: Der Vorgangsteuerung Teledienst zur Integration von Vorgangsteuerungen. In: Sandkuhl, K.; Weber, H. (Hrsg.): Telekooperationssysteme in dezentralen Organisationen. Tagungsband GI-Workshop 31/96, Berlin, 1996.
- [DGL00] Deiters, W.; Goesmann, T.; Löffeler, T.: Flexibility in Workflow Management: Dimensions and Solutions. In: International Journal of Computer Systems Science and Engineering - Special Issue on Flexible Workflow Technology Driving the Networked Economy, Vol. 15, Nr. 5, CRL Publishing Ltd, S. 303-313.
- [DGS95] Deiters, W.; Gruhn, V.; Striemer, R.: Der FUNSOFT-Ansatz zum integrierten Geschäftsprozeßmanagement. In: Wirtschaftsinformatik, 37(1995) 5, S. 459-466.
- [DGS98] Deiters, W.; Goesmann, T.; Striemer, R.: Risikogetriebene Vorgehensmodelle zur Entwicklung von Workflow-Management-Anwendungen. In: [HSW98b], S. 107-123.
- [DHMM96] Dourish, P.; Holmes, J.; MacLean, A.; Marquardsen, P.; Zbyslaw, A.: Freeflow: Mediating Between Representation and Action in Workflow Systems. In: Proceedings of the ACM Conference on Computer Supported Cooperative Work (CSCW'96). Cambridge, MA, 1996, S. 190-198.
- [DHMS00] Diefenbruch, M.; Hoffmann, M.; Misch, A.; Schneider, H.: Situated Knowledge Management. In: Reimer, U. (Hrsg): Proceedings of the Third International Conference on Practical Aspects of Knowledge Management (PAKM2000), 2000, <http://sunsite.informatik.rwth-aachen.de/Publications/CEUR-WS/Vol-34/>

- [DIN96] Deutsches Institut für Normung e.V.: Geschäftsprozeßmodellierung und Workflow-Management. Forschungs- und Entwicklungsbedarf im Rahmen der entwicklungsbegleitenden Normung (EBN). DIN-Fachbericht 50, Beuth Verlag, 1996.
- [DJB96] Davenport, T. H.; Jarvenpaa, S. L.; Beers, M. C.: Improving Knowledge Work Processes. In: Sloan Management Review, 34(4), 1996, S. 53-65.
- [DP98] Davenport, T. H.; Prusak, L.: Working Knowledge. How organizations manage what they know. Mc-Graw-Hill, 1998.
- [DVÖ96] Derungs, M.; Vogler, P.; Österle, H.: Metamodell Workflow, Hochschule St. Galler, Institut für Wirtschaftsinformatik, Bericht Nr. IM HSG/CC PSI/3, 1996.
- [EKR95] Ellis, C.; Keddara, K.; Rozenberg, G.: Dynamic Change Within Workflow Systems. In: Proceedings of the Conference on Organizational Computing Systems (COOCS'95), Milpitas, CA, 1995, S. 10-21.
- [Elli79] Ellis, C. A.: Information Control Nets: A Mathematical Model of Office Information Flow. In: Proceedings of the 1979 ACM Conference on Simulation, 1979.
- [Fais96] Faisst, W.: Wissensmanagement in Virtuellen Unternehmen. Arbeitspapier der Reihe „Informations- und Kommunikationssysteme als Gestaltungselement Virtueller Unternehmen“, Nr. 8/1996, Universität Bern, Universität Leipzig, Universität Erlangen-Nürnberg.
- [Flan99] Flanagan, D.: JAVA Enterprise in a nutshell. O'Reilly & Associates Inc., Sebastopol, 1999.
- [FS94] Ferstl, O.; Sinz, E. J.: Grundlagen der Wirtschaftsinformatik, 2. Aufl., München: Oldenbourg, 1994.
- [Fuhr97] Fuhr, N.: Information Retrieval - Skriptum zur Vorlesung. Universität Dortmund, 1997.
- [Gall95] Galler, J.: Metamodelle des Workflow-Managements. Universität des Saarlandes, Forschungsberichte des Instituts für Wirtschaftsinformatik an der Universität des Saarlandes, Heft 121, Saarbrücken, 1995.
- [GFS98] Goesmann, T.; Föcker, E.; Striemer, R.: Wissensmanagement zur Unterstützung der Gestaltung und Durchführung von Geschäftsprozessen. Bericht des Fraunhofer-Institut für Software- und Systemtechnik ISST, 48/98, Dortmund, 1998.
- [GH00] Goesmann, T.; Hoffmann, M.: Unterstützung wissensintensiver Geschäftsprozesse durch Workflow-Management-Systeme. In: Reichwald, R.; Schlichter, J. (Hrsg.): Verteiltes Arbeiten - Arbeit der Zukunft. Tagungsband der D-CSCW 2000. Leipzig et al.: Teubner, 2000, S. 139-152.
- [GH01] Goesmann, T.; Herrmann, Th.: Wissensmanagement und Geschäftsprozessunterstützung am Beispiel des Workflow Memory Information System WoMIS. In: [HSW01], S. 83-101.

- [GKLS98] Goesmann, T.; Krämer, K.; Löffeler, T.; Striemer, R.: Prototyping bei Workflow-Projekten: Entwicklung und Einführung einer Workflow-Management-Anwendung bei der DHL Worldwide Express GmbH. In: Herrmann, Th.; Just-Hahn, K. (Hrsg.): Groupware und organisatorische Innovation. Tagungsband der D-CSCW 1998, Stuttgart et al.: Teubner, S. 311-318.
- [GKSW98] Goesmann, T.; Krämer, K.; Striemer, R.; Wernsmann, C.: Ein Kriterienkatalog zur Bestimmung der Eignung von Workflow-Management-Technologie. In: [HSW98b], S. 95-106.
- [Goes00] Goesmann, T.: Supporting Knowledge Intensive Processes with Workflow Management Systems. In: Proceedings Workshop "Beyond Workflow Management - Supporting Dynamic Organizational Processes" of the ACM Conference on Computer Supported Cooperative Work (CSCW'2000). <http://www.stern.nyu.edu/~abernste/cscw2000/accept.htm>.
- [Goes01a] Goesmann, T.: KontextNavigator - Ein Organizational Memory zur Workflow-Unterstützung wissensintensiver Prozesse. In: Proceedings zum Workshop „Wissensmanagement und E-Business“ im Rahmen der Jahrestagung der Gesellschaft für Informatik, Wien, 2001, Band 1, S. 61-67.
- [Goes01b] Goesmann, T.: KontextNavigator - A Workflow Integrated Organizational Memory Information System to Support Knowledge-Intensive Processes. In: Proceedings zum „Second Workshop on Supporting Organizational Learning“ (SOL 2001), Tokyo, 2001.
- [GP00] Gross, T.; Prinz, W.: Gruppenwahrnehmung im Kontext. In: Reichwald, R.; Schlichter, J. (Hrsg.): Verteiltes Arbeiten - Arbeit der Zukunft. Tagungsband der D-CSCW 2000. Leipzig et al.: Teubner, 2000, S. 115-126.
- [GPP96] Gance, N. S.; Pagani, D. S.; Pareschi, R.: Generalized Process Structure Grammars (GPSG) for Flexible Representations of Work. In: Proceedings of the ACM Conference on Computer Supported Cooperative Work (CSCW'96). Cambridge, MA, 1996, S. 180-189.
- [GS98] Goesmann, T.; Striemer, R.: Entwicklung von Workflow-Management-Anwendungen in der Praxis - Erfahrungen und Konsequenzen. Bericht des Fraunhofer-Institut für Software- und Systemtechnik ISST, 44/98, Dortmund, 1998.
- [Habe99] Habermann, F.: Organizational-Memory-Systeme für das Management von Geschäftsprozessen. Forschungsberichte des Instituts für Wirtschaftsinformatik an der Universität des Saarlandes, Heft 154, Saarbrücken, 1999.
- [Hahn99] Hahn, C.: Konzeption eines History-Management-Systems im Anwendungsfall Workflow-Management. Diplomarbeit, Universität Erlangen-Nürnberg, 1999.
- [Heil94] Heilmann, H.: Workflow Management: Integration von Organisation und Informationsverarbeitung. In: HMD - Praxis der Wirtschaftsinformatik 31 (1994) 176, S. 8-21.

- [Herr95] Herrmann, Th.: Workflow Management Systems: Ensuring organizational Flexibility by Possibilities of Adaption and Negotiation. In: Proceedings of the ACM Conference on Organizational Computing Systems (COOCS'95), 1995, S. 83-95.
- [HGH98] Hoffmann, M.; Goesmann, T.; Herrmann, T.: Erhebung von Geschäftsprozessen bei der Einführung von Workflow-Management. In: [HSW98a], S. 15-72.
- [HGM01] Hoffmann, M.; Goesmann, T.; Misch, A.: Unsichtbar oder Vergessen - Wie man „verborgenen Wissensprozessen“ auf die Schliche kommt. In: [SSSS01], S.59-63 und in: [MHAM01].
- [HHJN99] Heintz, P.; Horn, S.; Jablonski, S.; Neeb, J.; Stein, K.; Teschke, M.: A Comprehensive Approach to Flexibility in Workflow Management Systems. In: Proceedings of the International Joint Conference on Work Activities Coordination and Collaboration WACC '99, San Francisco, 1999.
- [HJ99] Holsapple, C. W.; Joshi, K. D.: Description and Analysis of Existing Knowledge Management Frameworks. In: Proceedings of the 32nd Hawaii International Conference on System Sciences (HICSS), 1999.
- [HLSS99] Hull, R.; Llibat, F.; Simon, E.; Su, J.; Dong, G.; Kumar, B.; Zhou, G.: Declarative Workflows that Support Easy Modification and Dynamic Browsing. In: Proceedings of the International Joint Conference on Work Activities Coordination and Collaboration (WACC '99), San Francisco, 1999, S. 69-78.
- [HNT99] Hansen, M. T.; Nohria, N.; Tierney, T.: Wie managen Sie das Wissen in Ihrem Unternehmen? In: Harvard Business Manager, 5/1999, S. 85-96.
- [HSW98a] Herrmann, Th.; Scheer, A.-W.; Weber, H. (Hrsg.): Verbesserung von Geschäftsprozessen mit flexiblen Workflow-Management-Systemen. Band 1: Von der Erhebung zum Sollkonzept. Heidelberg et al.: Physica, 1998.
- [HSW98b] Herrmann, Th.; Scheer, A.-W.; Weber, H. (Hrsg.): Verbesserung von Geschäftsprozessen mit flexiblen Workflow-Management-Systemen. Band 2: Von der Sollkonzeptentwicklung zur Implementierung von Workflow-Management-Anwendungen. Heidelberg et al.: Physica, 1998.
- [HSW99] Herrmann, Th.; Scheer, A.-W.; Weber, H. (Hrsg.): Verbesserung von Geschäftsprozessen mit flexiblen Workflow-Management-Systemen. Band 3: Erfahrungen mit Implementierung, Probebetrieb und Nutzung von Workflow-Management-Anwendungen. Heidelberg et al.: Physica, 1999.
- [HSW01] Herrmann, Th.; Scheer, A.-W.; Weber, H. (Hrsg.): Verbesserung von Geschäftsprozessen mit flexiblen Workflow-Management-Systemen. Band 4: Workflow Management für die lernende Organisation - Einführung, Evaluierung und zukünftige Perspektiven. Heidelberg et al.: Physica, 2001.
- [Humm01] Fulcrum KnowledgeServer 3.5 - Technical Overview. Technical Paper. Hummingbird Ltd., 2001, <http://www.hummingbird.com>
- [Hype00] HIS User's Guide - Hyperwave Information Server Version 5.5. Hyperwave AG, 2000.

- [JB96] Jablonski, S.; Bußler, C.: Workflow Management - Modelling Concepts, Architecture and Implementation. London et al.: International Thomson Computer Press, 1996.
- [JBS97] Jablonski, S.; Böhm, M.; Schulze, W. (Hrsg.): Workflow Management: Entwicklung von Anwendungen und Systemen - Facetten einer neuen Technologie. Heidelberg: dpunkt-verlag, 1997.
- [JC99] Jorgensen, H. D.; Carlsen, S.: Emergent Workflow: Planning and Performance of Process Instances. In: Proceedings of the 1999 Workflow Management Conference Workflow-based Applications, Arbeitsbericht Nr. 70, Universität Münster, 1999, S. 98-116.
- [JHS98] Just-Hahn, K.; Hagemeyer, J.; Striemer, R.: Verbesserung von Geschäftsprozessen mit flexiblen Workflow-Management-Systemen: Ein Überblick über das Projekt MOVE. In: [HSW98a], S.1-13.
- [Joos95] Joosten, S.: Conceptual Theory for Workflow Management Support Systems. Centre for Telematics and Information Technology, University of Twente, 1995.
- [Jous] The Joust Outliner. <http://www.ivanpeters.com/>
- [JR99] Just-Hahn, K.; Rolles, R.: Integriertes Feedback als KVP-Werkzeug. In: [HSW99], S. 63-87.
- [KA98] Kühn, O.; Abecker, A.: Corporate Memories for Knowledge Management in Industrial Practice: Prospects and Challenges. In: [BP98], S. 183-206.
- [KDB98] Klein, M.; Dellarocsa, C.; Bernstein, A. (Hrsg.): Towards adaptive workflow systems. Workshop of the 1998 ACM Conference on Computer Supported Cooperative Work (CSCW'98), <http://ccs.mit.edu/klein/cscw98/>
- [Kidd94] Kidd, A.: The Marks are on the Knowledge Worker. Proceedings of CHI'94, S. 186-191.
- [KJSR99] van Kaathoven, R.; Jeusfeld, M. A.; Staudt, M.; Reimer, U.: Organizational Memory Supported Workflow Management. In: Scheer, A.-W.; Nüttgens, M. (Hrsg.): Electronic Business Engineering. 4. Internationale Tagung Wirtschaftsinformatik 1999. Heidelberg et al.: Physica, 1999, S. 543-563.
- [KT00] Karagiannis, D.; Telesko, R.: The EU-Project PROMOTE: A Process-oriented Approach for Knowledge Management. In: Reimer, U. (Hrsg): Proceedings of the Third International Conference on Practical Aspects of Knowledge Management (PAKM2000), 2000, <http://sunsite.informatik.rwth-aachen.de/Publications/CEUR-WS/Vol-34/>
- [KU94] Kirn, S.; Unland, R.: Workflow Management mit kooperativen Softwaresystemen. State of the art und Problemabriß. Technischer Bericht Institut für Wirtschaftsinformatik, Nr. 29, Universität Münster, 1994.
- [Kuhl99] Kuhlmann, A.: Entwurf und Einsatz eines Organizational Memory im Rahmen des Geschäftsprozeßmanagements. Diplomarbeit, Universität Dortmund, 1999.

- [Lehn00] Lehner, F.: Organisational Memory - Konzepte und Systeme für das organisatorische Lernen und das Wissensmanagement. München, Wien: Hanser-Verlag, 2000.
- [LGJM96] Lee, J.; Gruninger, M.; Jin, Y.; Malone, T.; Tate, A.; Yost, G. et al.: The Process Interchange Format and Framework, Working Group, Version 1.1, 1996.
- [LH98] Loser, K.-U.; Herrmann, Th.: Management von Wissen ist mehr als Management von Information. In: Computerwoche 9/1998, S. 97-100.
- [Lind99] Lindert, F.: Fraktales Prozeßmanagement. Dissertation, Technische Universität Berlin, 1999.
- [Liss01] Lissa, C.: Erweiterung des Informationsaspekts von Workflow-Management-Systemen. Diplomarbeit, Universität Dortmund, 2001.
- [LMK98] Lehner, F.; Maier, R.; Klosa, O.: Organizational Memory Systems. In: Reimer, U. (Hrsg.): Proceedings of the 2nd International Conference on Practical Aspects of Knowledge Management (PAKM98), <http://sunsite.informatik.rwth-aachen.de/Publications/CEUR-WS/Vol-13/>
- [Luca01] Lucas, R.: WIS Benutzeranleitung (Systemversion 2.0). Fraunhofer-Institut für Software- und Systemtechnik ISST, Dortmund, 2001.
- [LW00] Lucas, R.; Weber, T.: Skill-Management: Building block for project management with flexible teams. In: Barzdins, J. (Hrsg.): Databases and information systems. Fourth International Baltic Workshop, Baltic DB&IS 2000, Boston: Kluwer Academic Publishers, 2000, S.153-160.
- [MAHM01] Müller, H.-J.; Abecker, A.; Hinkelmann, K.; Maus, H. (Hrsg.): Geschäftsprozessorientiertes Wissensmanagement. Workshop im Rahmen der 1. Konferenz Professionelles Wissensmanagement. Forschungsbericht D-01-02, Deutsches Forschungszentrum für Künstliche Intelligenz DFKI, 2001 und <http://sunsite.informatik.rwth-aachen.de/Publications/CEUR-WS/Vol-37/>
- [Maus99] Maus, H.: Towards a Functional Integration of Document Analysis and Understanding in Workflow Management Systems. In: Proceedings of the 1999 Workflow Management Conference Workflow-based Applications, Arbeitsbericht Nr. 70, Universität Münster, 1999, S. 83-97.
- [MD99] Maurer, F.; Dellen, B.: A Concept for an Internet-based Process-Oriented Knowledge Management Environment. In: Proceedings of the Knowledge Aquisition Workshop (KAW98), Banff, Canada, 1998, <http://ksi.cpsc.ucalgary.ca/KAW/KAW98/KAW98Proc.html>
- [MDBG00] Maurer, F.; Dellen, B.; Bendeck, F.; Goldmann, S.; Holz, H.; Kötting, B.; Schaaf, M.: Merging Project Planning and Web-Enabled Dynamic Workflow Technologies. In: IEEE Internet Computing, Mai/Juni 2000, S. 65-74.
- [MH99] Maurer, F.; Holz, H.: Knowledge Management for Learning Software Organizations. Proceedings of KnowTech '99, Potsdam, 1999.
- [MRT91] Meghini, C.; Rabitti, F.; Thanos, C.: Conceptual modeling of multimedia documents. In: IEEE Computer, October 1991.

- [NT97] Nonaka, I.; Takeuchi, H.: Die Organisation des Wissens: Wie japanische Unternehmen eine brachliegende Ressource nutzbar machen. Frankfurt am Main, New York: Campus Verlag, 1997.
- [Nutt96] Nutt, G.: The evolution toward flexible workflow systems. In: Distributed Systems Engineering, No. 3, 1996, S. 276-294.
- [Oest98] Oestereich, B.: Objektorientierte Softwareentwicklung: Analyse und Design mit der Unified Modeling Language. 4. Auflage, München: Oldenbourg-Verlag, 1998.
- [Open99] Livelink 8 - Putting Knowledge to Work. Benutzerdokumentation. OpenText Corp., 1999.
- [ORG00] Ohlhausen, P.; Rüger, M.; Grote, P.: Wissensmanagement heute - Untersuchungsergebnisse bei deutschen Unternehmen. In: Wissensmanagement, 3/2000, S. 55-58.
- [Pare96] Pareschi, R.: Workflow Management and Knowledge Management - Summary of the Workshop on Adaptive Workflow held at the Conference on Practical Applications of Knowledge Management. In: Proceedings of the 1st International Conference on Practical Aspects of Knowledge Management (PAKM96), Zürich, 1996.
- [PB99] Perez, A. G.; Benjamins, V. R.: Overview of Knowledge Sharing and Reuse Components: Ontologies and Problem-Solving Methods. In: Proceedings of the IJCAI-99 Workshop on Ontologies and Problem-Solving Methods (KRR5), 1999, <http://sunsite.informatik.rwth-aachen.de/Publications/CEUR-WS/Vol-18/>
- [Pfen00] Pfennigschmidt, S.: Metadaten in der Informationslogistik - Modellierung von Informationsdiensten. Interner Bericht, Fraunhofer-Institut für Software- und Systemtechnik, Berlin, 2000.
- [Pola85] Polanyi, M.: Implizites Wissen. Frankfurt am Main: Suhrkamp, 1985.
- [PR87] Picot, A.; Reichwald, R.: Bürokommunikation. Leitsätze für Anwender. Hallbergmoos, 1987.
- [PR95] Picot, A.; Rohrbach, P.: Organisatorische Aspekte von Workflow-Management-Systemen. In: IM Information Management & Consulting, Nr. 1, 1995, S. 28-35.
- [PRR98] Probst, G. J. B.; Raub, S.; Romhardt, K.: Wissen managen: Wie Unternehmen ihre wertvollste Ressource optimal nutzen. 2. Auflage, Frankfurt am Main: Frankfurter Allgemeine Zeitung für Deutschland, Wiesbaden: Gabler, 1998.
- [Reim97] Reimer, U.: Knowledge Integration for Building Organisational Memories. In: Proceedings des Workshops "Knowledge-Based Systems for Knowledge Management in Enterprises" im Rahmen der 21st Annual German Conference on AI '97 (KI-Jahrestagung '97), Freiburg, 1997.
- [RL99] Remus, U.; Lehner, F.: The Role of Process-oriented Enterprise Modeling in Designing Process-oriented Knowledge Management Systems. In: Staab, S.; O'Leary, D. (Hrsg.): Bringing Knowledge to Business Processes. 1999 AAAI Spring Symposium, Technical Report SS-00-03, AAAI Press, 1999, S. 30-36.

- [RMGB98] Rey, M.; Maassen, A.; Gadeib, A.; Brücher, H.: Stufenmodell zur Einführung von Wissensmanagement. In: IM Information Management & Consulting, 1 (1998), S. 30-36.
- [RPR99] Rupprecht, C.; Peter, G.; Rose, T.: Ein modellgestützter Ansatz zur kontextspezifischen Individualisierung von Prozeßmodellen. In: Scheer, A.-W.; Nüttgens, M. (Hrsg.): Electronic Business Engineering. 4. Internationale Tagung Wirtschaftsinformatik 1999. Heidelberg et al.: Physica, 1999, S. 353-373.
- [RW73] Rittel, H.; Webber, M.: Dilemmas in a General Theory of Planning. In: Policy Sciences, 4 (1973), Elsevier, S. 155-169.
- [SGJR96] Sheth, A.; Georgakopoulos, D.; Joosten, S.; Rusinkiewicz, M.; Scacchi, W.; Wileden, J.; Wolf, A. (Hrsg.): Report from the NSF Workshop on Workflow and Process Automation in Information Systems. Technical Report UGA-CS-TR-96-003, Dept. of Computer Science, University of Georgia, 1996.
- [SGL98] Striemer, R.; Goesmann, T.; Löffeler, T.: Prototyping bei der Entwicklung und Einführung von Workflow-Management-Anwendungen. In: [HSW99], S. 113-134.
- [Shum98] Shum, S. B.: Negotiating the Construction and Reconstruction of Organizational Memories. In: [BP98], S. 55-78.
- [SHW97] Striemer, R.; Holten, R.; Weske, M.: Beschreibung und Analyse von Vorgehensmodellen zur Einführung von betrieblichen Workflow-Management-Anwendungen. In: Montenegro, S.; Kneuper, R.; Müller-Luschnat, G. (Hrsg.): Vorgehensmodelle - Einführung, betrieblicher Einsatz, Werkzeug-Unterstützung und Migration, Beiträge zum 4. Workshop, GMD-Studien Nr. 311, Sankt-Augustin: GMD-Forschungszentrum Informationstechnik, 1997.
- [SS99] Staab, S.; Schnurr, H.-P.: Knowledge and Business Processes: Approaching an Integration. In: Proceedings of the International Workshop on Knowledge Management and Organizational Memory (IJCAI '99), Stockholm, 1999.
- [SS00] Staab, S.; Schnurr, H.-P.: Smart Task Support through Proactive Access to Organizational Memory. In: Knowledge-Based Systems, 13(5), Elsevier, 2000, S. 251-260.
- [SSSS01] Schnurr, H.-P.; Staab, S.; Studer, R.; Stumme, G.; Sure, Y. (Hrsg.): Professionelles Wissensmanagement: Erfahrungen und Visionen. Beiträge der 1. Konferenz Professionelles Wissensmanagement. Aachen: Shaker-Verlag, 2001.
- [Stri98] Striemer, R.: Ein zielbasierter Ansatz für das Requirements Engineering modellbasierter Informations- und Kommunikationssysteme. Dissertation, Technische Universität Berlin, 1998.
- [SZ95] Stein, E. W.; Zwass, V.: Actualizing Organizational Memory with Information Technology. In: Information System Research 6(2), S. 85-117.
- [Theb00] TheBrainSDK - Software Development Kit Documentation. The Brain Technologies Corp. <http://www.thebrain.com/products/brainsdk/download/default.html>

- [Warg97] Wargitsch, C.: Ein Organizational-Memory basierter Ansatz für ein lernendes Workflow-Management-System. FORWISS-Report FR-1997-004, Bayerisches Forschungszentrum für Wissensbasierte Systeme FORWISS, Erlangen, 1997.
- [Warg98] Wargitsch, C.: Ein Beitrag zur Integration von Workflow- und Wissensmanagement unter besonderer Berücksichtigung komplexer Geschäftsprozesse. Dissertation, Universität Erlangen-Nürnberg, 1998.
- [WDA99] Wolf, T.; Decker, S.; Abecker, A.: Unterstützung des Wissensmanagements durch Informations- und Kommunikationstechnologie. In: Scheer, A.-W.; Nüttgens, M. (Hrsg.): Electronic Business Engineering. 4. Internationale Tagung Wirtschaftsinformatik 1999. Heidelberg et al.: Physica, 1999, S. 745-765.
- [Webe99] Weber, H.: Informationslogistik und Wissensmanagement. In: Proceedings KnowTech 1999, Potsdam, 1999.
- [WfMC95] Workflow Management Coalition: The Workflow Reference Model, Document-Number TC00-1003, 1995, <http://www.aiim.org/wfmc>
- [WfMC99a] Workflow Management Coalition: Terminology and Glossary. Document-Number WFMC-TC-1011, Version 3.0, 1999, <http://www.aiim.org/wfmc>
- [WfMC99b] Workflow Management Coalition: Interface 1: Process Definition Interchange Process Model. Document-Number WfMC-TC-1016-P, Version 1.1, 1999, <http://www.aiim.org/wfmc>
- [WGHS99] Weske, M.; Goesmann, T.; Holten, R.; Striemer, R.: A Reference Model for Workflow Application Development Processes. In: Proceedings of the International Joint Conference on Work Activities Coordination and Collaboration WACC '99, San Francisco, S. 1-10; ebenfalls in: Software Engineering Notes, Vol. 24, No. 2, 1999.
- [Wiig96] Wiig, K. M.: Knowledge Management is No Illusion. In: Wolf, M.; Reimer, U. (Hrsg.): Proceedings of the 1st International Conference on Practical Aspects of Knowledge Management (PAKM96), Zürich, 1996.
- [Wiig98] Wiig, K. M.: Perspectives on Introducing Enterprise Knowledge Management. In: Proceedings of the 2nd International Conference on Practical Aspects of Knowledge Management (PAKM98), <http://sunsite.informatik.rwth-aachen.de/Publications/CEUR-WS/Vol-13/>
- [WU91] Walsh, J. P.; Ungson, G. R.: Organizational Memory. In: Academy of Management Review, 16, 1 (1991), S. 57-91.
- [WWT98] Wargitsch, C.; Wewers, Th.; Theisinger, F.: An Organizational-Memory-Based Approach for an Evolutionary Workflow Management System - Concepts and Implementation. In: Proceedings of the 31st Hawaii International Conference on System Sciences (HICSS), 1998, Vol. I, S. 174-183.

Abbildungsverzeichnis

Abbildung 1: Referenzmodell der Workflow Management Coalition [WfMC95].....	12
Abbildung 2: Der Prozesslebenszyklus [JHS98]	14
Abbildung 3: Typen von Daten in WFMS [WfMC99a].....	17
Abbildung 4: Workdesk-Client mit Umlaufmappe bei CSE Workflow	19
Abbildung 5: Wissenspyramide nach [AN95] (Beispiel an Anlehnung an [PRR98])	24
Abbildung 6: Wissenslebenszyklus [NT97]	25
Abbildung 7: Drei Säulen des Wissensmanagements	27
Abbildung 8: Bausteine des Wissensmanagements [PRR98].....	28
Abbildung 9: Grundmodell des Information Retrieval [Fuhr97]	36
Abbildung 10: Sichten auf den Inhalt von Dokumenten (in Anlehnung an [Fuhr97]).....	37
Abbildung 11: Modell des Business Knowledge Management, dargestellt anhand eines Beispiels [BVÖ99]	40
Abbildung 12: Vorgehen zum Knowledge Process Redesign [Allw98].....	41
Abbildung 13: Doppelter Evolutionszyklus [Warg98]	44
Abbildung 14: Logischer Aufbau von WorkBrain [Warg98]	45
Abbildung 15: Referenzmodell für die Prozessmodellierung [JC99].....	47
Abbildung 16: Relevante Informationen zur Durchführung unstrukturierter Aktivitäten	48
Abbildung 17: Logische Architektur von KnowMore [ABS99]	51
Abbildung 18: Integration von Workflow und kontextabhängigen Sichten [SS00]	53
Abbildung 19: Integration von EULE2 in ein WFMS [KJSR99].....	54
Abbildung 20: Die PROMOTE-Methodologie [KT00].....	56
Abbildung 21: Spektrum von Prozessen zwischen Routineprozessen und Wissensarbeit	62
Abbildung 22: Wissenstransfer innerhalb und zwischen Geschäftsfällen	64
Abbildung 23: Zugriff auf Wissen bei Prozessen unterschiedlicher Ausführungshäufigkeit und Durchlaufzeit	69
Abbildung 24: Der Prozess "Contract Management"	72
Abbildung 25: Unterschiedliche Quellen prozessbezogenen Wissens	78
Abbildung 26: Beispiele für Typen prozessbezogenen Wissens	79
Abbildung 27: Strukturierung sekundären Wissens anhand der Workflow-Objekte	83
Abbildung 28: Typen und Instanzen von Kontexten und kontextbildenden Objekten.....	91
Abbildung 29: Ausschnitt aus dem Workflow-Metamodell der Workflow Management Coalition [WfMC99a, WfMC99b].....	94
Abbildung 30: Ausschnitt aus dem Workflow-Metamodell von Galler [Gall95]	95
Abbildung 31: Ausschnitt aus dem Metamodell des WFMS CSE Workflow.....	96
Abbildung 32: Kontextbildende Objekttypen	99

Abbildung 33: Exemplarische Kontexte zu einem durch Aktivität und Geschäftsfall festgelegten Kontexttyp.....	103
Abbildung 34: Klassen von Sichtbarkeitsbedingungen.....	115
Abbildung 35: Aktive Bereitstellung eines Kontextdokuments.....	125
Abbildung 36: Verwaltung von Kontexten innerhalb des Prozesslebenszyklus.....	132
Abbildung 37: Aktivitäten bei der Initialisierung von Kontexten.....	134
Abbildung 38: Hierarchie von Verantwortlichkeiten zur Wissensbewahrung in Kontexten .	137
Abbildung 39: Architektur eines erweiterten WFMS.....	151
Abbildung 40: Aufbau des Kontext-Clients (Beispiel).....	153
Abbildung 41: Darstellung des aktuellen Kontexts.....	155
Abbildung 42: Aufbau der Kontextdatenbank.....	161
Abbildung 43: Die Architektur des WFMS CSE Workflow.....	165
Abbildung 44: Darstellung von Kontexten und Kontextdokumenten in WoMIS.....	168
Abbildung 45: Ändern von Attributwerten von Kontextdokumenten in WoMIS.....	169
Abbildung 46: Integration von WoMIS in CSE WebDesk.....	170
Abbildung 47: Architektur von WoMIS.....	171
Abbildung 48: Darstellung von Kontexten und Kontextdokumenten in KontextNavigator..	174
Abbildung 49: Darstellung des aktuellen Kontexts in KontextNavigator.....	176
Abbildung 50: Kontextstruktur in KontextNavigator.....	177
Abbildung 51: Informationsmodell für Kontextdokumente in KontextNavigator.....	178
Abbildung 52: Systemdokument zur Anzeige der Häufigkeit der Durchführung einer Aktivität.....	180
Abbildung 53: Architektur von KontextNavigator.....	183
Abbildung 54: Aufbau der Kontextdatenbank von KontextNavigator.....	187

Tabellenverzeichnis

Tabelle 1: Merkmale expliziten und impliziten Wissens	25
Tabelle 2: Beispiele für primäres und sekundäres Wissen	81
Tabelle 3: Kontextbildende Objekttypen und beispielhafte Inhalte zugehöriger Kontexttypen	100
Tabelle 4: Übersicht über Kontexttypen und Regeln zur Anwendung	102
Tabelle 5: Attribute des Aspekts „Identifikation“	107
Tabelle 6: Standardtypen von Kontextdokumenten.....	108
Tabelle 7: Attribute des Aspekts „Inhalt“	109
Tabelle 8: Attribute des Aspekts „Lebenszyklus“	112
Tabelle 9: Attribute des Aspekts „Adressaten und Berechtigungen“	114
Tabelle 10: Mögliche Sichtbarkeitsbedingungen für Kontextdokumente in unterschiedlichen Kontexttypen	118
Tabelle 11: Zulässige Datentypen, Operatoren und Konstanten in Teilbedingungen	119
Tabelle 12: Attribute des Aspekts „Sichtbarkeit“	120
Tabelle 13: Ereignisse zur aktiven Bereitstellung von Kontextdokumenten	122
Tabelle 14: Ereignisse und Bedingungen zur aktiven Bereitstellung von Kontextdokumenten	124
Tabelle 15: Attribute des Aspekts „Bereitstellung“	126
Tabelle 16: Übersicht über die Funktionalitäten von WoMIS.....	167
Tabelle 17: Übersicht über die Funktionalitäten von KontextNavigator	173
Tabelle 18: Beschreibung der Datenbanktabellen.....	188

Index

A

Aktivität	10
Aktivitätsblock	93
aktueller Kontext	154, 176
Anforderungen	77
Anwendungsfallanalyse	148

B

Bausteine des Wissensmanagements	26
Bearbeiter	11
Bereitstellungsmodus	120
Bramble	47

C

Contract Management	70, 141
CSE Workflow	18
Architektur	165
Metamodell	95
WebDesk	166, 170

D

Dokument	10, 106
primäres	79
sekundäres	79
Dokumentenmodellierung	35
Dokumentverantwortlicher	138

E

Ereignis der Prozessbearbeitung	122
EULE2	53

G

Geschäftsfall	11
---------------------	----

I

Information Retrieval	36
Informationsmodell zu Kontextdokumenten	
Adressaten und Berechtigungen	112
Bereitstellung	120
Identifikation	105
Inhalt	107
Lebenszyklus	110
Sichtbarkeit	114

J

JAVA	
Applet	180
RMI-API	181
Servlet	181

K

KnowMore	49
Kontext	89
Evaluierung	140
Initialisierung	133
Nutzung	136
Visualisierung	148, 153, 175
kontextbildender Objekttyp	89, 100
kontextbildendes Objekt	89
Kontext-Client	152, 174
Kontextdatenbank	160, 186
Kontextdokument	90
Kontextdokumente	
Standardtypen	108
KontextNavigator	172
Kontext-Server	157
Kontexttyp	89
Kontextverantwortlicher	139

M

Milos	55
MOBILE	18

O

Ontobroker	51
Organizational Memory	29
Organizational Memory Information System (OMIS)	29
prozessbezogen	77

P

PROMOTE	56
Prozess	10
prozessbezogenes Wissen	77
Charakteristik	79
Strukturierung	81
Typ	79
Prozessgruppe	66
Prozesslebenszyklus	13, 86, 131
Prozessmodelldokument	11
Prozessverantwortlicher	140

R

Routineprozess	20, 62
----------------------	--------

S

Sichtbarkeitsbedingung	115, 179
dokumentbezogen	115
prozessbezogen	115
Systemarchitektur	150, 171, 179
Systemdokument	127, 162, 179, 186
Standardbibliothek	126

T

TheBrainSDK	182
-------------------	-----

V

Verbundprojekt MOVE	6, 70
---------------------------	-------

W

Werkzeug	94
Wicked Problems	34
Wiederholbarkeit	121
Wissen	24
explizites	24
implizites	24
primäres	80
prozessbezogen	77
sekundäres	80
Wissens(ver)teilung	27
Wissensarbeit	33, 62
Wissensaufbau	84
Wissensbewahrung	27, 67, 84
wissensintensive Aktivität	62
wissensintensiver Prozess	63, 70
Wissenslebenszyklus	25
Wissensmanagement	22
prozessbezogen	39
Wissenspyramide	24
Wissenstransfer	64
WoMIS	166
WorkBrain	43
Workflow	11
Variable	115
Workflow Management Coalition	12, 93
Workflow-Management-Anwendung	12
Workflow-Management-System	11
Workflow-Metamodell	92
Workflow-Objekt	83, 89
Workflow-Objekttyp	89
Workware	46